

Web 2.0 i Authentic Travel – prosesser, aktører og usecases



Alen Cemer

Institutt for Informatikk, ved Universitet i Oslo.

Høst 2006

Rapporten er levert som en del av studiet til mastergraden.

Veileder: Dino Karabeg

SAMMENDRAG

Web 2.0 er definisjon av en form for ny trend og retning internett utvikler seg i. Kort oppsummert så er det hele en sammenblanding av metoder for hvordan nettsider oppfører seg. Nettsider i form av tjenester som benytter fellesskapet for å gjøre innholdet rikere for andre lesere, hvorav leser er den som muligens får mest fokus i dette websamfunnet da den samtidig har muligheter til bl.a. å bli forfatter også.

Jeg lurte på hvordan denne sammenblanding av metoder for hvordan en nettsider oppfører seg, kan brukes på en nettside som omhandler reising og opplevelser av autentiske reisemål. Etter å ha søkt på Internett, så har jeg kommet over en hel masse nettstedet som beskriver reising og varierte reiseopplevelser, men svært få av disse nettsidene er satt opp på en måte som gjør det mulig for at folk som er glad i å reise og har rike reiseopplevelser, utvikler og skaper nettstedet med det de selv beskriver.

Samme mulighet vil åpnes for aktører som tilbyr og organiserer reiseopplevelser. Min besvarelse vil fokusere mest på turisme som omhandler autentiske reisemål, opplevelser og aktører som tilbyr dem. Tilbydere er små samfunn og foretak som har tatt internett i bruk for å profilere og markedsføre seg, men ikke på en slik måte at dagens teknologiske muligheter er tatt i bruk. De har websider som oppdateres med lange mellomrom, samt at sidene er nesten umulig å finne på internett ved hjelp av søkemotorer. En av grunnene til dette er mangel på ferdigheter til å oppdatere informasjonen og legge det ut på internett. Under utviklingen av systemet vil brukervennligheten bli satt i fokus, slik at alle med grunnleggende datakunnskaper skal finne systemet lett å bruke, og samtidig bruke det regelmessig. Reiseopplevelser skapes og endres daglig, så de ville vært svært nyttig å få disse nyhetene og oppdateringene snarest mulig.

Når en organisasjon eller foretak legger informasjonen på internett gjennom en webmaster eller en gruppe av dem, er det utvilsomt at informasjonen blir forsinket, samt at en del aldri publiseres. En del av min besvarelse vil være et nytt system for organisasjonen Authentic Travel og deres webbehov, samtidig vil systemet gjøre det mulig for folk flest å dele sine reiseopplevelser med andre, beskrive dem med tekst og bilder, legge inn relevante linker, diskutere og kommentere informasjonen. Jeg vil ta metoder som brukes i Web 2.0 for å imøtekomme dagens behov som folk har for egen eksponering, skape verdifull informasjon gjennom deltakelse, skape sosiale nettverk, sist men ikke minst skape et sted der tilbud og etterspørsel møtes.

INNHALDSFORTEGNELSE

SAMMENDRAG.....	III
INNHALDSFORTEGNELSE.....	V
TABELLER OG FIGURER	IX
TAKK TIL.....	XI
1 INTRODUKSJON	1
1.1 Definisjon av problemstilling	3
1.2 Tenkelig scenario.....	4
1.3 Min implementasjon.....	4
1.4 Oppgavens inndeling.....	4
2 BAKGRUNN	6
2.1 Interaktivitet.....	7
2.2 Sosialt	7
2.3 Teknologier som kjennetegner Web 2.0	8
2.4 Andre kjennetegn ved Web 2.0	10
2.5 Finansielt	11
2.6 Web 2.0 – for og imot	11
3 TEKNOLOGI.....	12
3.1 Evig (perpetual) Beta	12
3.2 Brukernettverkeffekter.....	14
3.3 Brukere som utviklere.....	17
3.4 Rikere brukeropplevelser	18
3.5 Tilpasset tilgang til innhold	19
4 FORRETNINGSPÅDRIVERE	20
4.1 Oppstartskapital.....	20

4.2	Forretningsmodeller.....	22
4.3	Annonseinntekter	23
4.4	Abonnentinntekter	24
4.5	Behov for utvidelse og dominans.....	24
4.6	Long Tail Marked	25
4.7	Exitstrategier og taksering	26
4.8	Web 2.0 vs. Bubble 2.0	27
4.9	Web 2.0 forretningsutfordringer	27
5	CASESTUDIER	28
5.1	Wikipedia	28
5.1.1	Teknologi.....	28
5.1.2	Forretningsmodell.....	28
5.1.3	Wikipedia i dag.....	29
5.2	Google Maps	30
5.2.1	Teknologi.....	30
5.2.2	Forretningsmodell.....	30
5.2.3	Google Maps i dag.....	30
5.3	del.icio.us	31
5.3.1	Teknologi.....	31
5.3.2	Forretningsmodell.....	31
5.3.3	del.icio.us i dag.....	31
5.4	flickr	32
5.4.1	Teknologi.....	32
5.4.2	Forretningsmodell.....	32
5.4.3	flickr i dag	32
6	METODER	34
6.1	Kort om informasjonssystemer	34
6.1.1	IS utvikling	34
6.1.2	Uviklingsmodeller	35
6.1.3	Modelleringsspråk	35
6.2	Autentisk reising.....	36
6.3	Prosjektoversikt.....	37
6.3.1	Prosjekt mål	37
6.3.2	Kravspesifikasjon	37
6.3.3	Delmål	38
6.3.4	Prosessmodell	39
6.4	Risikovurdering	40
6.4.1	Erfaringsmangel (R1)	40
6.4.2	Verktøyproblemer (R2)	40
6.4.3	Sykdom (R3)	40
6.4.4	Frafall (R4)	40

6.4.5	Ikke godkjente delmål (R5)	40
6.4.6	Missforståelser av kravspesifikasjonen (R6)	40
6.4.7	Hardwareproblemer (R7)	40
6.5	Tekniske konsekvenser	41
6.6	Usecase	42
6.6.1	Aktører.....	42
6.7	Funksjonalitet for aktører Traveler og Provider	43
6.7.1	Add/Edit Article	44
6.7.2	Add/Edit Link	45
6.7.3	Add/Edit Attachment.....	46
6.7.4	Comment Article/Photo	46
6.8	Funksjonalitet for aktøren Admin	47
6.8.1	Edit Article	48
6.8.2	Add/Edit Company (provider).....	48
6.8.3	Add/Edit Banner.....	49
6.8.4	Add/Edit Internal Link.....	49
6.9	Sub systemer	50
6.10	Detaljerte usecase beskrivelser.....	50
6.10.1	Add/Edit Article.....	50
6.10.2	Add/Edit Link	51
6.10.3	Add/Edit Attachment	52
6.10.4	Add/Edit Comment.....	53
6.10.5	Search Provider.....	54
6.10.6	Search Archive.....	55
6.10.7	Edit Article.....	56
6.10.8	Add/Edit Company (provider)	57
6.10.9	Add/Edit Banner	58
6.10.10	Add/Edit Internal Link.....	59
6.11	Plattform og utviklingsverktøy	60
6.11.1	ER diagram – Authentic Travel	61
6.11.2	ER diagram – Blogg	62
6.11.3	ER Diagram – Forum.....	63
6.12	Sidestruktur (3-nivå).....	64
6.13	SQL.....	65
6.13.1	Liste artikler.....	65
6.13.2	Liste land	65
6.13.3	Liste tjenester.....	65
6.13.4	Liste de 10 siste fra blogg	65
6.13.5	Liste de 10 siste fra forum	66
6.13.6	Liste artikler knyttet til et bestemt land	66
6.13.7	Liste eksterne linker (bokmerker)	66
6.13.8	Liste vedlegg (bilder).....	66
6.13.9	Eksterne tjenester	66
6.13.10	Vise artikkel.....	67
6.13.11	Liste kommentarer (artikkel)	67
6.13.12	Liste kommentarer (bilde)	67
6.13.13	Vise banner (annonse)	67
6.14	Forretningsmodell	68
6.15	Sosiale effekter.....	68

7	EKSPERIMENTER	69
8	VIDERE ARBEID	70
9	KONKLUSJON.....	71
	VEDLEGG.....	73
	STIKKORDREGISTER	76
	REFERANSER	78

TABELLER OG FIGURER

Tabell 1, O'Reilly's Web 2.0 vs. Web 1.0	6
Tabell 2, Delmål for prosjektet	38
Tabell 3, Risikovurdering	40
Tabell 4, Risikohåndtering	41
Tabell 5, Aktører	42
Tabell 6, Usecase beskrivelse – Add/Edit Article	50
Tabell 7, Usecase beskrivelse – Add/Edit Link	51
Tabell 8, Usecase beskrivelse – Add/Edit Link	52
Tabell 9, Usecase beskrivelse – Add/Edit Comment	53
Tabell 10, Usecase beskrivelse – Search Provider	54
Tabell 11, Usecase beskrivelse – Search Archive	55
Tabell 12, Usecase beskrivelse – Edit Article	56
Tabell 13, Usecase beskrivelse – Add/Edit Company (provider)	57
Tabell 14, Usecase beskrivelse – Add/Edit Banner	58
Tabell 15, Usecase beskrivelse – Add/Edit Internal Link	59
Figur 1, "Meme Map" av Web 2.0 som ble utviklet som en brainstorming på FOO Camp, en konferanse på O'Reilly Media. Illustrasjonen viser konsepter som er under utvikling. Disse stråler ut av en Web 2.0 kjernetankegang.	2
Figur 2, Illustrasjon av de nye Web 2.0 firmalogoene	10
Figur 3, GMail BETA logo	13
Figur 4, Metcalfe's lov får fram at verdien av systemet er lik antall brukere opphøyd i en annen.	14
Figur 5, Seks klassiske komponenter som inngår i en forretningsmodell	22
Figur 6, En illustrasjon av <i>Long Tail</i> konseptet	25
Figur 7, Reise – Kultur – Teknologi	36
Figur 8, Vannfallsmodell – prosessene strømmer etter hverandre	39
Figur 9, Usecase for Traveler og Provider	43
Figur 10, Brukergrensesnitt – Add/Edit Article	44
Figur 11, Brukergrensesnitt – Add/Edit Link	45
Figur 12, Brukergrensesnitt – Add/Edit Photo	46
Figur 13, Brukergrensesnitt – Add/Edit Comment	46
Figur 14, Usecase diagram for Admin	47
Figur 15, Edit Article	48
Figur 16, Brukergrensesnitt – Add/Edit Company (provider)	48
Figur 17, Brukergrensesnitt – Add/Edit Banner	49
Figur 18, Brukergrensesnitt – Add/Edit Internal Link	49
Figur 19, ER diagram – Authentic Travel	61
Figur 20, ER diagram – blogg	62
Figur 21, ER diagram – forum	63
Figur 22, Sidestruktur – 3 nivå	64

TAKK TIL

Jeg vil takke Dino Karabeg for hans veiledning gjennom problemstilling som sto foran meg. Jeg setter stor pris på hans tips som alltid har satt meg på riktig spor!

Jeg vil takke samtidig til alle sammen som har vært rundt meg gjennom studietiden min og støttet meg opp i det arbeidet jeg gjorde til enhver tid. Aller mest vil jeg takke meg selv og tålmodigheten jeg hele tiden har hatt for at jeg skal klare alle utfordringer som sto foran meg!

1 Introduksjon

World Wide Web startet som et lite nettverk mellom universiteter og forskningssentrene. Målet var å ha tilgang til dokumenter som disse institusjonene hadde bruk for. I dag er web et avansert interaktivt medium for utveksling av informasjon, ikke minst for også å drive virksomhet. Web har hatt en kontinuerlig utvikling gjennom alle år. Web 2.0 er en viktig del av den seneste utviklingen. Teknologier bak Web 2.0 muliggjør unike brukermulighetene, der brukeren skaper informasjonen, samt benytter seg av skapt informasjon på unike måter. Web 2.0 står for mye mer enn bare teknologi, rettene sagt en blanding av teknologier. Web 2.0 representerer sosiale interaksjonen, nye kommunikasjonsmåter, nye måter å drive forretning på, etc. Det er det første konsept som kombinerer tekniske, sosiale og forretningsteorier.

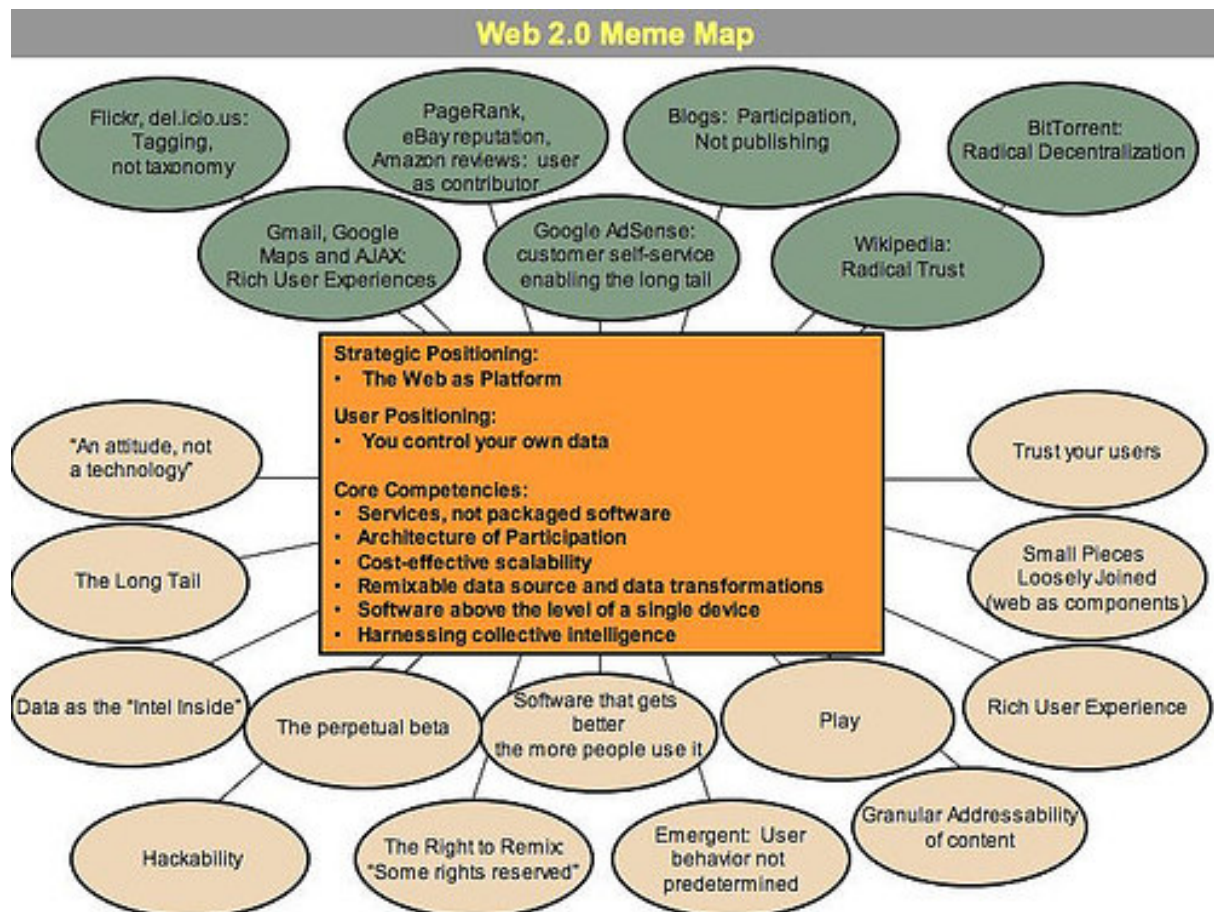
Som vi forstår av navnet symboliserer Web 2.0 noe nytt i forhold til hva vi har sett på weben tidligere. Men, hva er det nye og hva er det nytt i forhold til? Disse spørsmålene vil man kunne få ulike svar på. Det er avhengig av hvem man spør. Her er det ingen fasitsvar, men forståelsen av begrepet bygger rundt mange trender en ser i forhold til hvilken utvikling weben tar, både på kort og lengre sikt.

En måte å se på Web 2.0 er å betegne et nettsted med statiske websider som Web 1.0 og et nettsted med dynamisk genererte websider som Web 1.5. Et nettsted hvor vi i tillegg har høy funksjonalitet på klientsiden, kombinert med et rikt brukergrensesnitt vil da kunne kalles Web 2.0.

En annen måte å bruke begrepet Web 2.0 på er å si at dette er en ny måte for å drive virksomhet på Internett. Det som er nytt er at man inviterer brukerne til å være med å delta og skape felles verdier, fremfor at brukerne bare er noen som leser informasjonen, eller bruker enveistjenester som ligger tilgjengelig. Vanlig netthandel med rent kjøp og salg er et eksempel på en tradisjonell virksomhet på Internett.

Å la brukerne kommentere produktene slik man kan på for eksempel Amazon.com og mange andre nettbutikker er et steg i retning Web 2.0. Hvis man drar dette enda lenger lar man brukerne skape selve verdien på nettstedet gjennom at de selv tilbyr det som er til salgs. Dette gjelder for eksempel auksjoner som eBay og QXL, og formidlingstjenester som FINN. Det virkelig interessante med dette er at verdien på tjenesten stiger med antall brukere som benytter den. En formidlingstjeneste med få brukere har liten verdi både for de som skal tilby noe og de som skal kjøpe noe.

Det er også viktig å merke seg at sosialnettverkstjenester, altså tjenester der nettverket av folk er det viktige, ikke begrenser seg til tjenester hvor kjøp og salg er målet. Verdien kan være i å komme i kontakt med andre, som i venne- og dating tjenester. Det kan være å oppnå felles kunnskapsbase, som tilfelle med Wikipedia, å dele bilder på flickr eller videosnutter på YouTube, er andre eksempler på nettverkstjenester. Mulighetene for å skape verdi gjennom å bygge sosiale nettverk er store og det finnes flere og flere eksempler på nettstedene som bygger suksess på denne oppskriften.



Figur 1, "Meme Map" av Web 2.0 som ble utviklet som en brainstorming på FOO Camp, en konferanse på O'Reilly Media. Illustrasjonen viser konsepter som er under utvikling. Disse stråler ut av en Web 2.0 kjernetankegang.

På internett opererer med minst tre definisjoner av Web 2.0:

- Overgang fra websider der de framsto som informasjonsbeholdere – til kilder av innhold og funksjonalitet.
- Et sosialt fenomen som refererer til en ny tilnærming for utvikling og fordeling av innholdet på web. Det kjennetegnes med åpen kommunikasjon, desentralisering av myndighet, frihet til å dele og gjenbruke, og "marked som samtale".
- En endring av økonomiskverdi av web.

1.1 Definisjon av problemstilling

Autentisk reising har vist seg å være mye mer enn kilden til personlig tilfredsstillelse eller mulig profitabel forretning. Det er også blitt instrument for utveksling av kulturer og deres verdier. Gjennom spørreundersøkelser har det kommet fram at autentiske reiseopplevelser har virket inspirerende for de reisende, samt uformglemmelig. Det viser seg også at autentiske reiseopplevelser forbedrer mottakelse for- og forståelse av andre kulturer. Det dannes gjerne tilhørighet til stedene som inngikk i de reiseopplevelsene. Dette hjelper selvfølgelig til å bryte opp fordommer for andre kulturer, og åpne for kommunikasjon mellom den reisende og omgivelsene som besøkes. Autentiske reiseopplevelser skaper også bedre kommunikasjon de reisende imellom. Dette kan dreie seg om mennesker som er i familieband, forretningsfolk, eller bare venner og bekjente.

“I saw the people in our group change completely during those three days we were together: their eyes began to shine, their conversations deepened. As if some magic had been happening to us. Of course, we were seeing some pretty nice places in Norway. But that was obviously not the main thing that was going on.” – uttalelse av en deltaker på Authentic Travel studietur i Norge 2005.

Autentiske reiseopplevelser og turisme det skaper virker positivt også på økonomien og videreutvikling av lokalsamfunn. Reiseopplevelser blir på sitt mest unike når lokalfolk får selv mulighet til å vekke dem til live, og de reisende opplever alt det på lokalbefolkning sine premisser.

Så lenge vi kan huske, har menneskene reist for å lære om andre kulturer og for å utforske seg selv. Vi befinner oss i en tid der kulturer endres og omformes. Å stimulere veksten av autentiske reiseopplevelser er viktigere enn noen gang. Den stadig økende kommersielle globaliseringen truer alt det unike som ennå ikke er erobret. Autentisk reisevirksomhet er med på å understøtte *kulturell miljøvern*, samt bevaring og friskmelding av allerede skadede kulturreike områder.

Authentic Travel er en organisasjon som fronter denne bevegelsen og vil gjerne være et naturlig møtested for de reisende og de som tilbyr reiseopplevelser. Pr. i dag eksisterer det ingen infrastruktur på Authentic Travel sin webdomene, så oppgaven min går ut på skape noe helt nytt. Kravspesifikasjonen bygger ut fra dette, og den defineres som: *design og implementere et informasjonssystem basert på Web 2.0 tankegang, beskrive prosesser, samt definere usecases som skal skape kommunikasjon mellom de aktuelle aktørene.*

Jeg er blitt oppfordret om ta gjerne i bruk teknologier og sub systemer som allerede eksisterer på markedet og har vist seg godt mottatt av brukerne.

1.2 Tenkelig scenario

Utviklingen av systemet startet med tanken om å lage en portal som skal gjøre det mulig for alle mennesker i verden å beskrive sine autentiske reiseopplevelser og autentiske steder de har besøk. Det strider mot ingen regler hvis informasjonen er skrevet på subjektiv måte. Det er verdifullt når masse av slik informasjon samles på et sted og det hele framstår som en kunnskapsbase. Hovedpoenget og verdien er at informasjonen deles med andre folk, diskuteres og kommenteres. Alle som er interessert vil få tilgang til å benytte seg av systemet. Alle innlegg som skrives går direkte online. Siden all innhold på sidene vil skapes av frivillige, og innholdet struktureres automatisk, vil det bare være nødvendig med en person som drifter systemet. Driften vil innebære oppdateringer av systemet, introduksjon av ny funksjonalitet, samt moderat vedlikehold. De første inntektene kan genereres gjennom annonser som foretak eller organisasjoner kan sette opp på informasjonssystemet.

1.3 Min implementasjon

Jeg ønsker å lage et webbasert system for organisasjonen *Authentic Travel*. Den endelige løsningen skal basere seg på moduler som allerede er utviklet av andre, samt skreddersydd og nyutviklet programvare som skal sørge for at hele løsning framstår i en sammenheng. På denne måten vil jeg bruke elektroniske tjenester som fungerer bra i dag og nyutviklet løsning som supplerer disse. Den skreddersydde løsningen vil ha den primære rollen i systemet. Det ferdige systemet skal kunne brukes som en offisiell webside for www.authentictravel.org. Om dette blir gjort mens vurdering av denne oppgaven pågår, er jeg ikke helt sikker på. Derfor, se: <http://folk.uio.no/alence/> Registrer deg som ny bruker og test ut applikasjonen. Hvis administrator kontrollpanel skal brukes, logg inn med brukernavn: *test*, og brukeropassord: *test*.

1.4 Oppgavens inndeling

Besvarelsen består av ni kapitler. Det første kapitlet gjør oss kjent med Web 2.0 som er et hovedutgangspunkt for måten oppgaven er besvart på. Dette kapitlet definerer problemstilling og skisserer en løsning på den.

Det andre kapitlet beskriver dypere Web 2.0 og går inn i aspektene som omhandler bl.a. det sosiale, interaktivitet, økonomi og andre kjennetegn ved Web 2.0

Det tredje kapitlet beskriver teknologi som ligger under de fleste Web 2.0 applikasjoner og systemer. En del av teknologier har blitt veldig populære å bruke etter at de har oppnådd stor suksess i forbindelsen med Web 2.0.

Det fjerde kapitlet beskriver de økonomiske aspektene ved Web 2.0 bedrifter og prosjekter. Det viser seg at økonomien ikke spiller en sentral rolle for utviklingene av Web 2.0 applikasjoner. Hva er da drivkraften bak utviklingen?

Det femte kapitlet beskriver casestudier som jeg har tatt av fire kjente Web 2.0 applikasjoner på markedet i dag. Kapitlet handler om Wikipedia, Google Maps, del.icio.us og flickr. Jeg har beskrevet teknologi som særpreger hver av dem, samt forretningsmodeller de baserer seg på og situasjonen de befinner seg i pr. dags dato.

Det sjette kapitlet er ganske omfattende og det beskriver den løsningene jeg har kommet med for å imøtekomme de kravene for systemet som skal tjene organisasjonen Authentic Travel og brukerne av systemet deres.

Det syvende kapitlet vil omhandle undersøkelser og tester jeg har gjort. Disse går ut på å finne ut hva testbrukerne synes om systemet. Testene vil gå ut på å sette robusthet og pålitelighet til systemet. Disse skal utføres ved hjelp av testbrukere samt programvare egnet for dette formålet.

Det åttende kapitlet vil omhandle videre arbeid rundt dette prosjektet. Jeg vil komme med en beskrivelse av teknisk funksjonalitet som kan implementeres med tiden, samt annet som kan bli gjort med systemet.

Hele besvarelsen vil da avsluttes med en konklusjon over det arbeidet jeg har gjort, samt lærdom som kan hentes og brukes på liknende prosjekter.

2 Bakgrunn

Web 2.0 er et uttrykk introdusert av O'Reilly Media i 2004. Web 2.0 har siden da vært en referanse til andregenerasjons internettbaserte tjenester. Denne nye generasjonen av tjenester lar brukere samarbeide og dele informasjonen online på mange nye måter. Eksempler på dette er sosialnettverksider¹ (social networking sites), wikis², internettfora³, blogger⁴, ol.

I åpningstalen på konferansen da Tim O'Reilly⁵ og John Battelle⁶ introduserte Web 2.0-tankegang, brukte de bl.a. nøkkelbegreper for Web 2.0 applikasjonen:

- Web som plattform
- Tjenester
- Kostnadsoverkommelig
- Data som drivkraft
- Netterverkeffekt realisert gjennom arkitektur for deltakelse
- Innovasjon gjennom forsamling

Businessområde	Web 1.0	Web 2.0
Annonsering	DoubleClick	Google AdSense
Bildedeling	Ofoto	Flickr
Fildeling	Akamai	BitTorrent
Musikkdeling	mp3.com	Napster
Encyclopedia	Britannica Online	Wikipedia
Personlig uttrykk	Personlige websider	Blogging
Organisering	Evite	upcoming.org and EVDB
Internett	Domenenavn spekulasjoner	søkemotor optimalisering
Trafikkmåling	sidevisninger	Kostnad pr. klikk
Innholdsutvikling	Publisering av webmasteren	Skape informasjon ved hjelp av brukerdeltagelse
Innholdsredigering	CMS (content management systems)	Wikis
Innholdsorganisering	Kataloger (taxonomy)	tagging ("folksonomy")
Motorikk	Utholdenhet	Følelse av tilhørighet

Tabell 1, O'Reilly's Web 2.0 vs. Web 1.0

2.1 Interaktivitet

Det finnes mange subjektive oppfatninger om av hva Web 2.0 handler om. Det som er felles for alle oppfatninger er først og fremst ideen om interaktivitet og sosiale måter å bruke nettet på. Et nett som er mer interaktivt enn i dag, et les-skriv nett. Innføring av Web 2.0-begrepet har også samtidig satt en ny betegnelse på det nettet slik vi har vært vant til fram til nylig – *Web 1.0*: Aviser, media hus, firmaer, privat personer og alle andre legger ut informasjon i forskjellige slags former som nettbrukerne konsumerer ved å lese, se på og høre på. Web 1.0 kjennetegnes altså ved at de fleste er passive mottagere av det de måtte ønske å formidle. Slik trend har oppstått og blitt varende til nå nylig da informasjonspublisering på nettet har blitt forbeholdt dem som hadde infrastruktur og kompetanse for å legge ut informasjonen online.

Det revolusjonerende med Web 2.0 er at det handler om at nettbrukerne går over fra å bare motta informasjon, til å selv å være en formidler av informasjon. Nye teknologier alene, samt teknologier som brukes i felleskap har gjort det mulig å designe web som åpner mulighet for banebrytende interaktivitet. Dette har latt seg gjøre med alle deler av den multimediale weben vi kjenner fra før. Eksempler på dette er tekst i form av Wiki⁷, bilder⁸, film⁹ (videoblogging), eller lyd¹⁰ (podcasts).

2.2 Sosialt

Det nye har gjort det enklere for nettbrukerne å formidle informasjon av forskjellige slag. Samtidig har den nye teknologien gjort det mulig for at sosial interaksjon oppstår på en helt annen måte enn tidligere. Den sosiale delen av Web 1.0 gikk omtrent bare ut på å snakke med andre over IM¹¹, sende e-post og delta i forumer. I Web 2.0 kommenterer og linker bloggere seg til hverandre, nettbrukerne kommenterer hverandres bilder på flickr⁸ og oppretter grupper hvor de samler bilder om, diskuterer, bestemte temaer og lignende. Nettbrukerne har gjennom dette fått sosiale nettverk, sosiale bokmerker¹², sosiale nyheter¹³, osv. Nettbrukerne forteller andre nettbrukere om hvilke nyheter de syns er viktig ved å sende det inn til, eller ved å stemme på det dem gjennom Digg¹³ f. eks, deretter diskuterer de nyheten sammen. Feeds lar leserene holde deg oppdatert på andres blogger, bokmerker, videoer, Digg-nyheter, to-do lister, osv. Ved å assosiere *tags* (nøkkelord) med bilder, bloggposter, videoer, podcasts, osv. gjør de det enklere for andre å finne fram til relevant informasjon. Tags er en veldig viktig del av både teknologien bak, og den sosiale delen av Web 2.0.

2.3 Teknologier som kjennetegner Web 2.0

Konseptene som er beskrevet ovenfor er den viktigste siden ved Web 2.0. Konseptene baserer seg på mange teknologier som ligger under overflaten og som gjør alt dette mulig. Disse underliggende teknologiene bør også nevnes fordi det er viktige deler av Web 2.0-bølgen som ruller over nettet for tiden. Det ble nevnt for eksempel feeds, og selv om dette er en fantastisk nyttig og enkel teknologi som de aller fleste blogger benytter seg av og de aller fleste nettlesere har innebygd støtte for, så er det langt fra alle som vet hva feeds er for noe. Under vil det bli på en lite teknisk måte beskrevet hva er de viktigste teknologiene som driver Web 2.0 framover:

- **Feeds:** Feeds gjør det mulig for nettbrukerne å følge med på blogger, nettaviser, nye bilder på Flickr, osv. uten å måtte hoppe fra nettside til nettside. Hver gang noen skriver noe nytt i bloggen sin blir bloggens feed oppdatert, og hver gang noen skriver en ny kommentar blir feeden for kommentarer oppdatert. Med en feedreader¹⁴ kan brukerne abonnere på feeds fra de sidene du ønsker å følge med på og på den måten få automatisk beskjed hver gang de oppdateres. Når man abonnerer på en podcast i iTunes¹⁵ eller NRK er det en feed som gjør slik at den laster ned nye podcasts automatisk. Ikonet Feed Icon indikerer at en side har en feed. Liten beskrivelse av de tekniske detaljene er at en feed er en XML-fil. Det finnes flere typer feeds. Fire versjoner av RSS og to versjoner av Atom. I tillegg er det mulig å lage en fil med en liste over feeds, men dette er ikke særlig vanlig enda. En slik fil heter OPML.
- **API/Mashups:** Et kjent eksempel på en Web 2.0 tjeneste/applikasjon er Google Maps. Men, Google Maps¹⁶ er jo ikke noe man kan bruke til å formidle informasjon eller være sosial på noen måte i seg selv. Det er her åpne API kommer inn. Fordi Google Maps tilbyr en åpen API kan hvem som helst å trekke ut informasjon og bruke det til å lage sine egne tjenester. En slik tjeneste kalles en mashup, men mashups kan like godt trekke ut informasjon fra feeds istedenfor en API. En mashup kan for eksempel la brukeren legge til notater på forskjellige punkter på Google Maps-kartet og så dele dette med andre, hente nyheter fra forskjellige kilder, eller vise andre, på Google Maps-kartet, hvor i USA det bor registrerte seksualforbrytere for eksempel
- **Ajax:** Ajax er egentlig ikke en ny teknologi, men snarere en sammensmelting av forskjellige eldre teknologier som JavaScript og XML. Ajax gjør blant annet at ny informasjon kan lastes inn på en side uten at hele siden må lastes på nytt, at du kan dra og slippe ting på en nettside på samme måte som i et program, og at brukervennligheten generelt kan bli veldig mye bedre enn vi er vant med fra Web 1.0. Ajax (sammen med utbredelsen av bredbånd) har banet veien for såkalte Web Apps (webapplikasjoner), applikasjoner som kjører på en server isteden for på den lokale datamaskinen. Et kjent eksempel på en webapplikasjon er Writely¹⁷, et webbasert tekstbehandlingsprogram som på mange måte ligner på det velkjente Microsoft Word, eller OpenOffice.org's Writer. Men Writely lar brukeren også lagre tekstdokumenter på nettet, skrive på et dokument sammen med andre i "real time", og enkelt dele dem med andre.
- **Wiki:** Wiki-teknologien daterer seg tilbake til 1995, lenge før noen hadde hørt om Web 2.0. I tillegg er Wiki ikke like mye brukt i Web 2.0-tjenester som de

ovennevnte teknologiene, men Wiki er like fullt en viktig del av Web 2.0. En av de eldste, og mest kjente, Web 2.0 tjenestene på nettet er Wikipedia. Wikipedia er et leksikon som er fullt og helt bygd opp rundt Wiki systemet. En Wiki gjør det mulig for hvem som helst å redigere eksisterende sider, og legge til nye sider, uten at forandringene må bli sett igjennom av en moderator først. Ofte trenger brukeren heller ikke registrere deg for å legge til din informasjon og kunnskap om det temaet som skrives om. Wikier er en enkel måte å sette opp en sosial kunnskapsdatabase på. Den beste Wiki man kommet over, utenom Wikipedia, er Knowmore.org, en oversikt over nyttig informasjon om alle mulige selskaper. Fokuset ligger på menneskerettigheter, miljø og etikk.

2.4 Andre kjennetegn ved Web 2.0

Foruten de sosiale og interaktive sidene ved de fleste Web 2.0 tjenester, samt bruk av de forskjellige teknologier nevnt ovenfor er det flere mindre viktige kjennetegn som går igjen i mange Web 2.0 firmaer. Disse inkluderer:

- **Spesielle navn:** Veldig mange Web 2.0 firmaer har navn som er litt annerledes. Dette inkluderer Meebo, Sproutit, Kaboodle ShoZu og mange flere (se Figur 2, Illustrasjon av de nye Web 2.0 firmalogoene).
- **Coming soon/e-postinnsamling:** Om brukeren går til nettsiden til en helt ny Web 2.0 startup vil du for det meste ikke se noe annet enn en side som sier at noe veldig spennende er på vei, og at om du vil få beskjed når dette spennende blir tilgjengelig så er det bare å gi dem e-postadressen din. Denne perioden kalles gjerne “stealth mode”.
- **Invitasjonsbasert:** Men når brukeren først får vite at tjenesten er tilgjengelig, så er det likevel ikke sikkert at den er tilgjengelig for den brukeren. Mange Web 2.0-tjenester er nemlig invitasjonsbaserte (mest sannsynlig etter inspirasjon fra Gmail), og om brukeren ikke gav dem e-postadressen sin tidlig får den kanskje ikke noen invitasjon. Denne praksisen brukes for å kunne teste ut hvordan vanlige folk reagerer på tjenesten slik at kritiske ting kan fikses på før man åpner den opp for alle.
- **Beta:** Men aller viktigst er ordet “beta”. De fleste Web 2.0-tjenester hevder at de er i beta selv lenge etter at invitasjonsfasen er over. Med beta menes at tjenesten enda ikke er fullt utviklet hva angår funksjonalitet, og at det kan oppstå problemer. Denne praksisen er sannsynligvis også inspirert av Google, som er beryktet for å holde tjenester i beta i mange år.



Figur 2, Illustrasjon av de nye Web 2.0 firmalogoene¹⁸

2.5 Finansielt

De fleste Web 2.0 tjenester og webapplikasjoner blir utviklet av små “startups”. Dette er helt nye enheter, gjerne bare bestående av mellom to og fem personer. Dette er gruppe gründere som har fått en god idé og begynner å utvikle den på egenhånd. Etter hvert får noen utvalgte startups økonomisk støtte fra investorer. Og noen av startups har etter hvert tjent inn store penger ved å bli kjøpt opp av gigantiske aktører i bransjen. Spesielt Yahoo! har vært aggressiv når det gjelder å kjøpe opp små Web 2.0-firmaer. Denne trenden er blitt fulgt opp av andre også. Disse er for eksempel Google og Fox (News Corp.). Store firmaer har åpnet lommeboka for å få en bit av Web 2.0-kaka. Disse gigantene har i liten grad selv utviklet egne Web 2.0-tjenester. Dette gjelder egentlig ikke Google med sine tjenester. Et annet unntak er også Microsoft som har jobbet aktivt med sin Windows og Office Live. Det enorme antall nye startups og tjenester, samt de store summene som enkelte ganger har vært inni bildet, har imidlertid gjort en del skeptiske til Web 2.0. Noen sammenligner det hele med dotcom-boblen som sprakk på slutten av nittitallet, og kaller det bare for Bubble 2.0.

2.6 Web 2.0 – for og imot

Selve begrepet *Web 2.0* har også skapt masse oppstyr. Flere blogger har dukket opp, ene og alene med den hensikt å rakke ned på begrepet og de som bruker det. Ett eksempel er Go Flock Yourself, som dukket opp kort tid etter at den Firefox-baserte nettleseren Flock ble lansert. Nettleseren som er spesielt beregnet på aktive brukere av Web 2.0. Men også mer seriøs kritikk av Web 2.0 har kommet fra mange hold. Blant annet Russell Shaw, en blogger på det anerkjente nettstedet ZDNet, har kommet med krass kritikk av Web 2.0. Shaw peker på at det ikke finnes noen felles standard som folk går etter. Teknologien og kjennetegnene ved Web 2.0 som er beskrevet ovenfor finnes helt klart, sier han, men det er ikke en enhetlig oppgradering av nettet. Det er bare en helt vanlig utvikling av ny teknologi og nye tjenester, slik som det alltid har vært, mener Shaw, og konkluderer med at Web 2.0 rett og slett ikke eksisterer. Dette fikk også den kjente Web 2.0-bloggen Read/WriteWeb til å erklære at Web 2.0 er død. Likevel nekter begrepet “Web 2.0” å dø, og stadig nye tjenester og Web Apps som setter en Web 2.0-merkelapp på seg selv dukker opp. Og nylig gikk dette bildet av en rekke Web 2.0 firmalogoer sin seiersgang rundt i bloggøsferen (se Figur 2, Illustrasjon av de nye Web 2.0 firmalogoene).

3 Teknologi

Dette kapitlet vil omhandle de nøkkelkonseptene som er relatert til Web 2.0. Som nevnt i introduksjonen, handler Web 2.0 mye om forretningsmodeller, sosiale interaksjon og teknologi. I visse miljøer påstås det at den store forskjellen mellom Web 1.0 og dagens web er reduksjon av betydningen det tekniske har under utviklingen som pågår. Mange av standarder, f. eks XML¹⁹, SOAP²⁰, samt infrastrukturer som Linux²¹ og Apache²² som brukes i Web 2.0 applikasjoner har blitt tatt for gitt, selv om de er godt etablerte standarder på markedet.

I dette kapitlet vil det bli satt fokus på tekniske aspekter som viser seg å ha sentrale roller av mange Web 2.0 sider, applikasjoner og diskusjoner. Det blir framstilt en forklaring bak ideer og prinsipper bak hver teknologi. Det blir også gjort et forsøk å framstille eksempler på sider og applikasjoner som bygger på teknologi som en sentral del av deres forretningsmodell. Det må sies at det er ikke bare disse teknologiene som er relevante til Web 2.0, heller ikke er det noen av websidene eller applikasjonene som baserer seg bare på en teknologi som kommer til å bli nevnt. Dette er heller ett sett av viktige konsepter som er en fellesnevner for mange Web 2.0 servicetilbud.

3.1 Evig (perpetual) Beta

Ideen bak evig beta er å slippe applikasjonen så tidlig som mulig ut til brukerne. Dette kan ofte være helt ubearbeidede prototyper som skal håve inn tilbakemelding fra brukerne på tidligst mulig fase av utviklingen. Det er også mening å måle begeistring for ideen, og tidligst mulig vise sin tilstedeværelse i et marked som vokser fort. Hvis det er banebrytende applikasjon det er snakk om, så er det viktig å vise sin tilstedeværelse i et marked som kommer til å vokse fort. Denne ideen likner på Raymond's²³ "Release Early, Release Often" artikkel der han argumenterer for rask iterativ applikasjonsutvikling med fokus på å lytte til brukernes tilbakemelding etter hver iterasjon.

I den tradisjonelle litteraturen som beskriver applikasjonsutvikling, står det at betatesting referer til testfasen gjort av de kommende brukerne av applikasjonen. Betatestene blir gjort først etter at alle testene er gjort av utviklingsteamet og applikasjonen ansees å være mer eller mindre klar for distribusjon. I Web 2.0 kontekst betegnes betatesting nærmest som den eneste testingen som applikasjonen går gjennom. Visse utviklingsmiljøer anser betaversjonen til å være den tilstand applikasjonen befinner seg hele tiden. Det er her "evig" (perpetual) betegnelsen av *perpetual beta* kommer fra. Denne praksisen har blitt kritisert²⁴, ikke minst for at den strider mot tradisjon innen applikasjonsutvikling.

Et eksempel på evig beta applikasjon er Google's GMail²⁵ elektronisk post tjeneste. GMail har vært tilgjengelig siden tidlig 2004, og nå teller flere millioner brukere. Applikasjonen har hatt utallige versjonsutgivelser. Hver av dem har blitt fikset for feil, og ikke minst introdusert ny funksjonalitet. Disse utgivelser har hele tiden vært tilgjengelig for brukerne, der har blitt informert over det som er blitt gjort og introdusert så de gjerne kan teste det ut og kommer med tilbakemelding. "BETA" tekst som er en synlig del av GMail logo, minner brukerne om hvilken status applikasjonen er i og ber om forbehold hvis feil oppdages.



Figur 3, GMail BETA logo

Det er noen underliggende antagelser angående evig beta applikasjoner. Det er som regel slik at brukerne betaler ikke for bruken av Web 2.0 applikasjoner, i alle fall ikke i den tidlige fasen av applikasjonen. Dette fører med seg at brukerne ikke bryr seg om mangel- og feilfull applikasjon²⁶. Dette er i sterk kontrast til de fleste IT bedriftene som satser på aktiv teknisk support i tilfelle noe går alt med applikasjonen. Web 2.0 bedrifter opererer typisk med en e-post adresse der brukerne kan sende sine klager, rapporter om feil, behov for funksjonalitet, etc. Det finnes ingen garanti for at slike henvendelser skal føre til at noe blir utbedret og gjort. Brukerne kan bare håpe på det beste.

En annen antagelse om evig beta er redusert forpliktelse fra utviklerens side. Siden applikasjonen deres sees som uferdig, hjelper dette utviklere og bedriften deres å redusere sannsynlighet for at de er skyldige hvis applikasjonen deres skulle gjort noe dramatisk overfor brukeren. I en mulig straffeprosess mellom partene, vil bedriften trolig ikke bli straffet noe særlig pga. betatilstand applikasjonen befinner seg i og betingelsene brukeren godtar når applikasjon er i slik tilstand. I det lange løpet kan dette skade bedriften når brukerne innser hva de kan bli utsatt for, og faktisk slutter å bruke applikasjonen.

Evig beta er derfor en hel prosess, i stede for en tilstand. Denne prosessen i seg selv er ikke ny. Microsoft og andre firmaer har gjennom en lang årrekke hatt betatesting strategier med eksterne brukere. Den nye aspekten med Web 2.0 er omfanget av denne prosessen. Dette går ut på hvor mange brukere er involvert i betatestingen, og varigheten av prosessen.

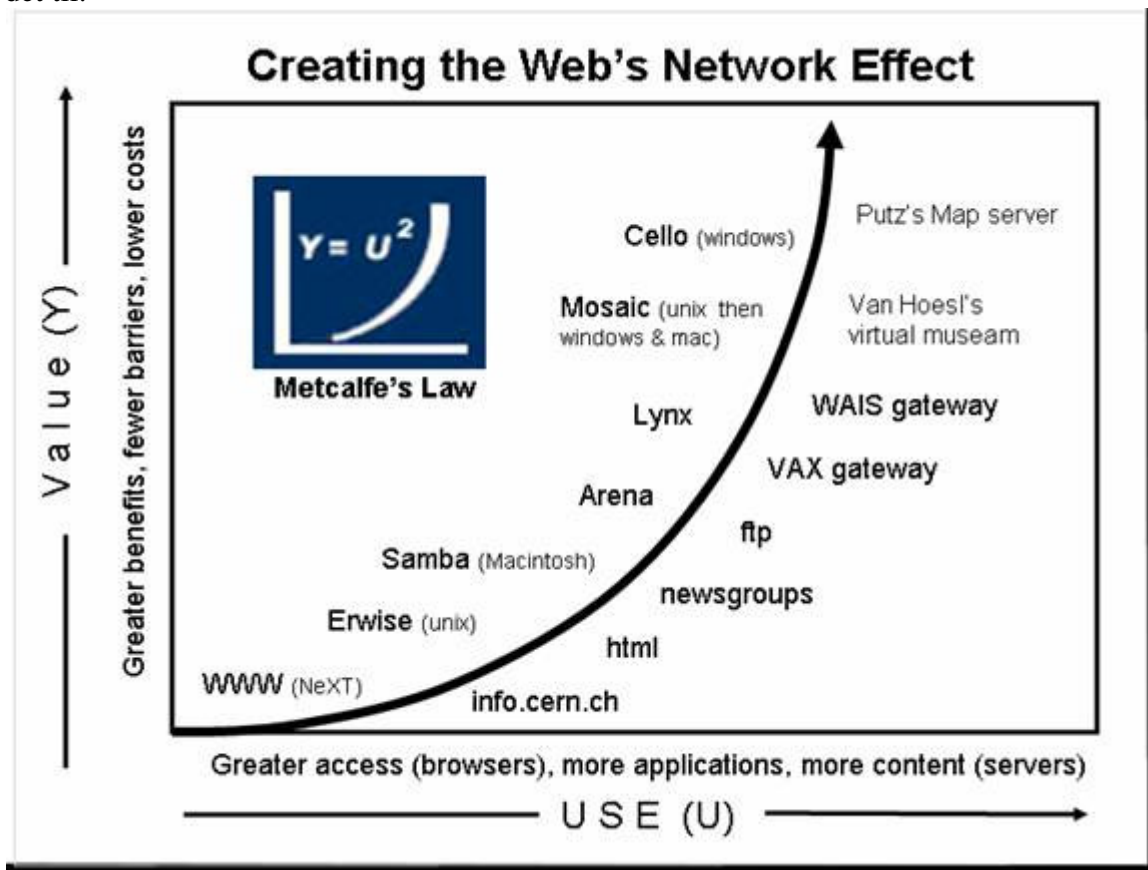
Det er lett å konkludere med at betafasen Web 2.0 applikasjonene befinner seg i er god pga. den ubetalelige prisen som brukeren gjør, der de for gjengjeld får en applikasjon som er tilpasset deres egne behov. Denne prosessen har også negative bivirkninger for erfarne applikasjonsutviklere, prosjektledelse, en hel rekke andre brukere som kan være like mye verdt, om ikke mer, enn de tidligste mottagere av applikasjonen.

3.2 Brukernetverkeffekter

En annen sentral aspekt av Web 2.0 websider og applikasjoner er det at de stoler på en hel masse brukere for at verdiene på skal skapes. Dette kan relateres til tillitt gitt til brukernetverkeffekter²⁷, eller sideeffektene som brukernetverkeffekter skaper.

Tradisjonelle applikasjoner trenger et brukervolum for å skape profitt, men verdien av applikasjonen er like stor for alle sammen uansett hvor mange brukere applikasjonen har. F. eks, Microsoft Word er like nyttig for en bruker som for en annen, eller en hel million andre brukere. Produktet av en Microsoft Word går alltid ut på det samme – å skape tekstdokumenter for brukeren, som senere kanskje skal være verdifulle for andre. Denne oppgaven er et eksempel på det.

Metcalfe's lov kalles fenomenet der verdien av et system er proporsjonell med antallet brukere som bruker systemet. Web 2.0 applikasjoner og websider baserer seg mye på denne loven, selv om denne loven ikke kan tolkes som universelt som mange skulle ha det til.



Figur 4, Metcalfe's lov får fram at verdien av systemet er lik antall brukere opphøyde i en annen.

En Web 2.0 applikasjon har typisk veldig liten verdi hvis det er få brukere som omgir den og aktivt interagerer med den. F. eks Wikipedia, som litt lengre ned i oppgaven vil bli beskrevet, ville ikke vært så nyttig og verdifull hvis det hadde vært få brukere som skriver i den. Verdien av Wikipedia kommer fra de mange tusener brukere som bidrar med personlig informasjon og synspunkter. Det er alle disse kombinerte bidrag som øker verdien av Wikipedia.

Behovet for mange brukere er en stor utfordring for Web 2.0 applikasjoner. Til i dag har det blitt brukt ”munn – til – munn” markedsføringsteknikk, der det primært gjennom populære blogger, online tidsskrifter og news har blitt allmenkjent at applikasjonen eksisterer, og der det er blitt gjort å nysgjerriggjøre brukere til å klikke seg innom. Eksempel på et slikt markedsføringsnettsted er Slashdot²⁸.

Det finnes tekniske utfordringer når en webside eller applikasjon har mange besøkende brukere. Det stilles tekniske krav til infrastrukturen som ligger under, som må tåle høy brukeraktivitet og at kostnadene for driften av infrastrukturen, dvs. båndbredde, lagringsplassen, databasen og liknende ikke setter en stopper for systemet. Disse utfordringene har med tiden blitt mindre og mindre da teknologien som inngår i en slik infrastruktur er blitt prisgunstig. Linux webserver, flere gibabyte databasekapasitet og ubegrenset båndbredde for datatrafikk, koster i dag under 10\$ pr. måned²⁹. Dette skaper en lav-kostnad, lav-risiko miljø som gjør det mulig å eksperimentere mer systemet, da det ikke er stor fare for økonomien og at det skal gå mye utover investert kapital.

Det at infrastrukturen er prisgunstig, har det ikke noe å gjøre med det at god applikasjonsutvikling trenger gode og kompetente utviklere. Gründere av Web 2.0 bedrifter sitter gjerne selv på en høykompetanse, men de er likeså avhengig av å knytte til seg et større teknisk team. Utfordringer med ansettelsesforhold, personale og andre ting på dette planet er like som de tradisjonelle bedrifter møter i hverdagen. Web 2.0 bedrifter har en trendy rykte på seg og menneskene som jobber i dem er ambisiøse. Dette gjør det velig attraktivt for kompetente utviklere å ville bli involvert i slike bedrifter. Web 2.0 bedrifter rekrutterer enkelt mennesker som vil involvere seg i prosjekter, gjerne for mindre lønn enn andre tradisjonelle IT bedrifter gir for den samme kompetansen. I gjengjeld for lavere lønn tilbyr Web 2.0 bedrifter aksjer og opsjoner, slik at profitt blir stor hvis prosjektet lykkes og begynner å tjene de store pengene.

Det mest utbredte kjennetegnet som er med å bygge opp et brukernettsverk er ”tagging”. Dette betegner en teknologi som tillater brukere av en Web 2.0 applikasjon å sette merkelapper (i praksisen bare beskrivende ord) på innholdet av applikasjonen og beskrive sin tokning av det de opplever i applikasjonen. På Flickr, nettstedet for deling av bilder, brukere kan sette tagger på sine egne bilder, samt andre sine. Bilder med tilnærmet like tagger vil framvises som relevante til hverandre. Dette blir nøkkelordene som brukerne benytter som verktøy for å finne ønskelig informasjon. F. eks bruker kan søke etter bilder merket med ”Norway” for å finne bilder relevant til dette søkekriteriet. Det er da mulig å se på mange bilder fra forskjellige brukere som omhandler ”Norway”, samt at det er mulig å se de andre bildene hver og en bruker har satt ut for visning. På del.icio.us, setter brukerne taggene på bokmerkene (URL-ene) som andre har registrert. Begge disse tilfellene er mer beskrevet senere. Det er også mulig for brukerne å abonnere på nyheter ved hjelp av RSS³⁰ feeds som vil da elektronisk gi beskjed hver gang det er lagt inn et nytt bilde med de taggene som brukeren overvåker. Likedan er det f. eks mulig for en vilkårlig bedrift å abonnere på en RSS reed som gir dem beskjed hver gang noen legger inn URL til bedriften deres inn i del.icio.us. Dette ville ikke vært noe særlig verdifullt hvis ikke Flickr og del.icio.us hadde hatt millioner av brukere.

Fra et teknisk perspektiv så kan vi se at infrastrukturen for tagging ikke er noe særlig ny eller kompleks. Taggene er typisk lagret i en separert implementert tabell som er en del av relasjonsdatabase, og som hører sammen med de andre tabellene som er en del av den underliggende infrastrukturen. F. eks, en del.icio.us databaseinnlegg (registrering) er

lagret i en tabell og denne har selvfølgelig sin egne unike ID. En annen tabell inneholder alle tagger som er lagt inn av brukerne til en bestemt bokmerke databaseinnlegg. Bokmerke ID og bruker ID blir brukt som fremmednøkler. Disse teksttaggene binder dokumentene i mellom, binder seg selv til det opprinnelige dokumentet, binder brukeren til dokumentene og taggene, dokumentene til brukeren, etc. Alt dette er informasjon som vises på websiden og er verdifull.

Investorer har fått med seg de mulighetene tagging åpner for, og ikke minst for bedrifter som del.icio.us med deres forretningsmodeller. Det er forståelig å tenke at den primære utfordringen og selve verdien ligger i den store massen av brukere og de nettverkeffektene brukerne skaper – og ikke pengene. Forretningsmodeller som Web 2.0 bedrifter opererer med vil trolig i framtiden skape store utfordringer for Web 2.0 bedriftene hvis ikke de åpner for mer tradisjonelle markedsmetoder og salgsteknikker, eller retter fokus på strategiske allianser og samarbeid med andre bedrifter og organisasjoner som vil generere enda flere brukere.

3.3 Brukere som utviklere

En annen viktig del av Web 2.0 applikasjoner er eksponering for allmenne applikasjon programmeringsrensesnitt (API). Mens tradisjonelle bedrifter ofte stiller ut API-ene til sine samarbeidspartnere i håp om de kan supplere funksjonalitet eller forbedre det endelige produktet, går Web 2.0 bedrifter en annerledes vei og stiller ut API-ene til allmennheten og brukermassen. Web 2.0 bedrifter betror seg da til allmennheten og de kommende brukerne som da signaliserer hvilke mulige brukerscenarioer applikasjonene kan håndtere. Disse kan da være scenarioer som i den opprinnelige bruksbeskrivelsen ikke var ment. Dette setter prosjektene i fare for å miste den økonomiske støtte de fikk i utgangspunktet da ikke lenger er de prosjektene de en gang var da investorene investerte pengene i dem.

Det er to ettervirkninger av det å stille ut API for allmennheten. Den ene er å skape interesser hos utviklingsmiljøer som er veloverveid da de er en spesiell type brukere, og håpe på av de ikke bare begynner å bruke applikasjonen, men også begynner å profilere den i sine egne blogger, forum og websidene. Ikke minst er dette viktig fordi de prater om applikasjonen innad i sine respektive omgangskretser. Fordelen med dette er at flere typer brukere av alle typer fra alle de forskjellige miljøer fatter interesser for applikasjonen. Dette er da med å sette i gang oppbygningene av det brukernettverket og effektene av den. Den andre ettervirkningen av å stille ut API er å stimulere utvikling av tilleggsapplikasjoner som baserer seg på applikasjonens data og API. Hvis noen av de tilleggsapplikasjonene lykkes, så vil det få positive ringvirkninger for den opprinnelige applikasjonen og den underliggende teknologien.

Et eksempel på utstilling av API til allmennheten er Google Maps API³¹. Google utviklet deres Maps tjeneste med tanke på at det skal være tilgjengelig via enkel API, og de stilte ut API til allmennheten. Alle kan lese og laste ned relevant dokumentasjon, samt kodeeksempler. Alle har rettighet til bruke API for å utvikle en ny webside eller applikasjon. Google fraskriver seg garantien at API hele tiden skal være slik den var på et gitt tidspunkt, eller at den i det hele tatt skal alltid være tilgjengelig. Men, siden Google Maps har en rolle som kraftfull pådriver av bruker- og utviklingsinnovasjonen, virker som om fjerning av Google Maps ikke akkurat er i nærmeste framtid. Tjenesten har blitt brukt av utallige brukere, som ikke sjelden har designet en vellykket applikasjon på basisen av Google Maps API. Dette har ført til at de samme menneskene er blitt fulltidsansatte i Google.

Hvis man ser seg bakover så er det lett å finne prosjekter med en åpen API. Dokumentasjon er som regel automatisk generert fra den skrevne koden med kommentarer og denne stilles ut tilgjengelig for dem som måtte være interessert. Bedriftene som vanligvis stiller ut API jobber typisk med webservices. Dette gjør det lettere å lære seg aksessere, analysere systematisk, og ikke minst vedlikeholde når det er nødvendig. Webservices bruker som regel SOAP (Simple Object Access Protocol) beskjerer kodet i XML format, men også andre formater som JavaScript (brukt av Google Maps). Webservices og allmenntilgjengelige API er to nøkkelbegreper når vi tenker på Web 2.0.

3.4 Rikere brukeropplevelser

Rikere brukererfaringer er en sentral del av Web 2.0 tankegangen. Med "rikere" menes det i denne sammenhengen to forskjellige meninger. Den ene meningen er bedre designet, mer intuitive, og mer brukervennlig grafisk brukergrensesnitt av websidene og applikasjonene. Den andre meningen går ut på et renere brukergrensesnitt, dvs. et brukergrensesnitt som ikke er overdrevent fylt med linker og andre elementer. Mange mener at Web 1.0 var et skritt bakover når det gjelder brukergrensesnitt, fra de applikasjonene og systemene IT brukere var kjent med. Overgangen fra velutviklede brukergrensesnitt og miljøer fra Apple og MS Windows med ikonbasert navigasjon til HTML sider der alt nærmest var bare ren tekst. Web begynte å utvikle seg til å bli en plattform for datainput, flerdelte grafiske brukergrensesnitt, visualisering av inputdata, etc. Web 1.0 ble en besværlig og ineffektiv da brukergrensesnitt for slike handlinger ble for primitivt.

Dette behovet satte i gang utvikling av en rekke teknologier forbundet med Web 2.0, der målet er å kunne realisere rikere brukeropplevelser. En teknologi har peker seg spesielt ut, og den heter AJAX (Asynchronous JavaScript And XML). Dette er en teknologi som gjør det mulig å utføre dynamisk endring og oppdatering av deler av en webside uten å behøve en fullstendig oppdatering fra serveren. Effekten av dette er en webside eller applikasjon som oppdater data og grafiske elementer uten en synlig forsinkelse, som ville bli tydelig synlig på en typisk Web 1.0 applikasjon. Et godt eksempel på en Web 2.0 applikasjon er Gmail.

AJAX som teknologi er utfordringer for å overta markedet. Dette skyldes behovet for utviklere som behersker XML, vanlig serverside applikasjonsdesign og programmering, samt JavaScript programmering. Nettlesere som er på markedet støtter AJAX i variert grad. Det finnes ingen relevante standarder, og utviklingsmiljøer har ennå ikke kommet på banen ordentlig. Mens alle disse tekniske finesser utarbeides, forventes det av utviklere å venne seg til AJAX bruk. Dette vil øke omslutning og interessen for denne teknologien, der effekten av dette vil bli at flere og flere websider og applikasjoner tar i bruk AJAX i større eller mindre grad.

Det finnes noen andre teknologier som ment å gjøre samme formålet som AJAX. Disse er litt annerledes hvis man ser på den tekniske biten av dem. De konkurrerer mot hverandre om markedsandeler, mens begge baserer seg på Macromedia's Flash spiller³². Denne er installert på mer enn 90 % av datamaskiner verden rundt. Den teknologien heter Flex³³ og er et serversideprodukt med et ledende utviklingsmiljø. Flex gjør det mulig for designere å utvikle websider i tilnærmet ren grafisk miljø, så konvertere websiden til en animasjon som kjører ved hjelp av Flash. Laszlo³⁴ er en annen teknologi som direkte konkurrerer mot Flex og har tilnærmet lik designprosess hvis man ser det fra teknisk ståsted. Et eksempel på Flexdrevet applikasjon er Sherwin-Williams' Color Visualizer³⁵.

Nøkkelord for å akseptere disse overnevnte teknologier er mulighetene for at ikke-programmere på egen hånd er i stand til å designe rike brukergrensesnitt. Disse behovene har eksistert når man også tenker Web 1.0 websider og applikasjoner. Utviklingsmiljøer som støtter behovene for utvikling følger som regel tett etter at nye teknologier er lansert. Disse verktøyene er en nøkkelfaktor for videre Web 2.0 utvikling.

3.5 Tilpasset tilgang til innhold

Teknologien som også knyttes til Web 2.0 kan beskrives som ”tilpasset tilgang til innhold”. Dette refererer til mulighetene brukerne har for å få tilgang til bare deler av informasjonen fra en applikasjon når de trenger dem. Et moteksempel til hva denne teknologien går ut på er Microsoft Word. Brukeren må starte hele applikasjonen og åpne hele dokumentet selv om brukeren er bare ute etter en liten del av informasjonen som ligger i dokumentet. Det hadde vært mye mer lettvinnt å kunne åpne en rask applikasjon som gjør det enkelt og raskt finne fram til den delen av teksten brukeren er ute etter. Dette er funksjonalitet mange Web 2.0 applikasjoner tilbyr.

En typisk teknikk som brukes for å få tilpasset tilgang til innhold er brukertilpasset RSS feeds. F. eks, Google News tilbyr mer enn bare forhåndsskannet RSS feeds for kategoriene som sport eller underholdning, men den tjenesten tilbyr også sette inn søkekriterier som brukes innen den valgte kategorien, slik at brukeren bare får filtrert informasjon rett i sin mailboks ettersom informasjonen blir tilgjengelig.

Slik funksjonalitet er forbundet med utfordringer knyttet til design og omfang av informasjonen. I det overnevnte ekseplet er det millioner av brukere som alle har sine individuelle behov og skreddersydde RSS feeds. En stor database er nødvendig for å lagre feeddefinisjoner, samt stor CPU kapasitet for å behandle feedoppdateringer. Denne utfordringen er stadig lettere å løse da kostnadene til nødvendig infrastruktur ikke er høye.

4 Forretningspådrivere

Dette kapitlet handler om forskjellige forretningsmodeller og finansielle pådrivere bak eksistering av Web 2.0 bedrifter og prosjekter de realiserer. Web 2.0 bedrifter er et helt nytt felt. Relativt få bedrifter har blitt etablert og enda færre er framgangsrrike. Nå for tiden opplever vi bare den første bølgen av Web 2.0 bedrifter og prosjekter. Forretningsmodellene som disse baserer seg på vil utvilsomt bli forbedret gjennom den kommende tiden ettersom det dannes mer kompetanse rundt feltet Web 2.0.

4.1 Oppstartskapital

Joe Kraus, the founder of Excite writes on his blog, "Excite.com took \$3,000,000 to get from idea to launch. JotSpot³⁶ took \$100,000".

I forhold til oppstart Web 1.0 bedrifter gjennomgikk på 90-tallet, kapitalbehovet og finansiering av Web 2.0 bedrifter er veldig annerledes. Dette gjelder spesielt oppstartsfasen. I de siste ti årene, har det forekommet en del dramatiske endringer av faktorer som påvirker kostnaden av en webbedrifts oppstart:

- **Hardware:** prisene på alle nye hardwarekomponenter fortsetter å synke. Bruken av brukte hardwarekomponenter har økt siden kvalitet og tilgang til dem har økt betydelig gjennom de siste årene. Auksjonswebsidene som eBay og QXL gjør tilgangen på brukte komponenter enkel.
- **Software:** Linux, Apache, MySQL, PHP og andre fritt tilgjengelige *Open Source*³⁷ teknologier har dramatisk redusert kostnadene for å få utviklet effektive webbaserte applikasjoner. Bedriftene trenger bare å bruke pengene på å utvikle funksjonalitet og programvare som understøtter det spesielle ved den forretningsvirksomheten bedriften driver med.
- **Infrastruktur:** Den økningen av webhotelltilbydere, fasiliteter, kapasiteten og båndbredden har redusert driftskostnader for bedrifter som tilbyr webbaserte applikasjoner og tjenester. Alt dette har for ikke så lang tid siden vært store utgiftsposter for de samme bedriftene.
- **Talent:** Tilgang til den globale arbeidskraften, høyt utdannede fagfolk og erfarne eksperter, har senket ned kostnadene forbundet med programvare arkitekter, designere og utviklere. Innleid arbeidskraft som sitter i helt andre verdensdeler og krever mindre betaling for tjenestene de leverer, er blitt en populær innsparingsmetode.
- **Markedsføring:** Web 2.0 bedrifter markedsfører seg primært over Internett. Strategier de bruker for å skape kjennskap til deres produkter er blogger, fora, websider, søkemotormarkedsføring, etc. Strategier som multimillion dyre *Super Bowl* annonser er ikke aktuelle for den nye generasjon av bedrifter. Webmarkedsføring er mer presis og kan rettes mot den brukergruppen som bedriften ønsker. Utgiftene en slik markedsføringsform bringer med seg baserer seg på antall besøkende/klikk og ikke bare stedet annonsen ligger på.

Resultat av de overnevnte faktorer er at Web 2.0 oppstarter ikke trenger å basere seg på pengekapital i en tradisjonell forstand. I stede baserer de seg på personlig kapital, venner og familie for å starte bedrifter. Når man går ut ifra at investorkapital ikke er nødvendig

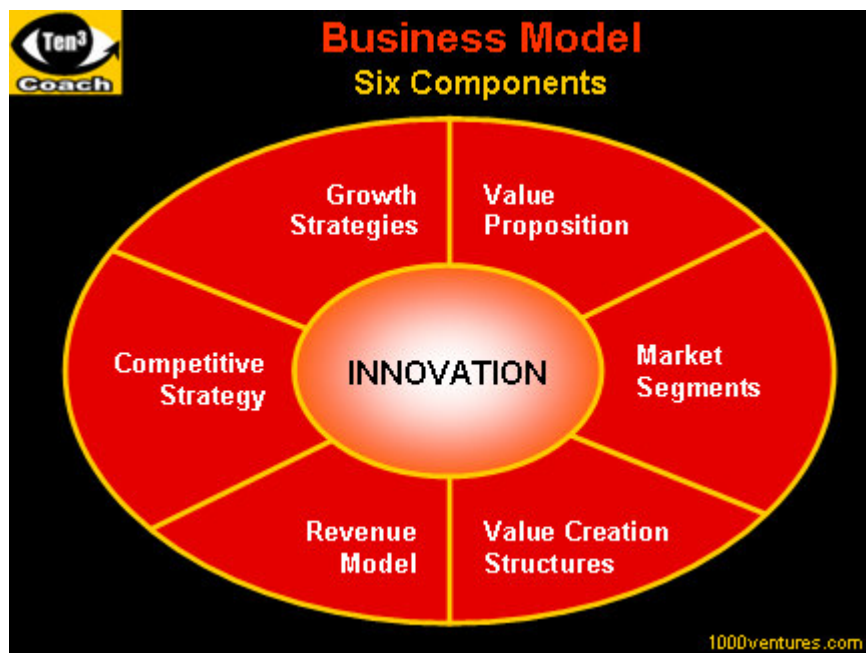
for denne typen bedrifter, dannes det unike egenskaper ved disse bedriftene i forhold til de tradisjonelle bedrifter med investorer som styrer bedrifter i bakgrunnen:

- Web 2.0 bedrifter etableres ofte av programvareutviklere som kan gjøre mesteparten av utviklingen selv eller med lite nettverk av partnere.
- Denne type gründere retter mesteparten av energien på utviklingen av løsningen i stedet for å sitte i forholdsvis lange perioder og utvikle forretningsplan.
- Gründere jobber ofte med den nye bedriften eller prosjektet i fritiden, uten å forlate deres faste jobb. Dette sikrer dem økonomisk stabilitet og i teorien betyr dette at utviklingen kan vare i en ubestemt tid.

4.2 Forretningsmodeller

Mange av de ferske Web 2.0 bedrifter som fortsatt regnes å være i oppstartfasen, står overfor de samme utfordringer som deres forgjengere på 90-tallet: mangel på klart definerte inntektsmuligheter. På lik linje med Web 1.0 bedrifter, en betydelig andel av Web 2.0 bedrifter som er i oppstartsfasen har ikke godt utarbeidet inntektsmodell. På en Web 2.0 begivenhet som ble avholdt på Cambridge i Massachusetts, tre bedrifter som er i oppstartsfasen har presentert seg selv og det de kommer til å tilby. Ikke en av dem kom med noen klare formeninger om hvordan de skal generere inntekt ut av ideen de skal realisere. En person som var tilstedeværende stilte et direkte spørsmål om akkurat det: "Hvordan har dere tenkt å tjene penger?" Svar på dette spørsmålet var uklart og tydelig ikke godt gjennomtenkt på forhånd: "Kanskje annonseinntekter eller noe liknende.". Det kom klart frem at utarbeidelser av meningsfulle inntektsstrategier ikke sto på prioritetslister til disse bedriftene.

En annen utfordring som er relatert til generering av inntekter for Web 2.0 bedrifter er at de ofte utleder deres verdier fra allerede etablerte datatjenester. Eksempel på dette er websiden *chicagocrime.org*³⁸ som kombinerer data fra Chicago Police Department med Google Maps. Denne og liknende applikasjoner er veldig nyttige til brukeren, men det er uklart hvordan inntektene fordeles mellom de forskjellige aktørene som deltar i den totale løsningen. Det er kanskje derfor Google, Yahoo! og andre som tilbyr API tilgang til deres elektroniske tjenester, setter en begrensning for antall spørringer som vil bli akseptert fra en slik kombinert løsning og understreker at data de leverer ikke kan brukes til kommersielle formål.



Figur 5, Seks klassiske komponenter som inngår i en forretningsmodell

4.3 Annonseinntekter

Mange Web 2.0 bedrifter som er i oppstartsfasen åpner mulighetene for annonseinntekter som en mulig løsning for å hente penger ut av de produktene eller tjenestene de tilbyr. Mye av denne interesse for annonseinntekter kan knyttes til relativt effektive annonseplattformer som tilbys av f. eks Google og Yahoo!. Google er for tiden den mest prominente tilbyder av denne typen tjenester, med sin veldig populære AdSense³⁹ konsept. AdSense gjør det mulig for den eller dem som driver websiden å vise Google-annonser på en eller flere av deres sider. Annonssørene betaler Google for hver klikk som er blitt foretatt via annonsene. Google i sin tur deler prosentandel av inntekten med websideeieren der annonsene ble utplassert.

Annonsering som inntektskilden via f. eks AdSense kanaler, er veldig fristende for Web 2.0 bedrifter i oppstartsfasen. Det tar veldig lite tid og innsats for å starte et samarbeid med Google, og begynne å tilby annonse på websidene sine. Muligheten for annonseinntekter gjør det mulig for bedriftene å tjene penger uten i første omgang å måtte ta betalt for de tjenestene og produktene de egentlig tilbyr. Meningen er å tjene penger på den trafikken som websiden deres genererer.

Annonseinntekter er en rask og praktisk måte å tjene penger på. Tross dette så er det en del risikoer som knytter seg til bedrifter som baserer sin inntekt bare på denne inntjeningsmetoden:

- Det er fremdeles begrenset antall aktører som binder annonsører og websideeiere. For tiden er Google den største aktøren på markedet. Mangel på en reell konkurranse gjør at makten sitter i hendene på de få etablerte aktørene.
- Annonseinntekter baserer seg primært på trafikken websiden genererer. Det finnes prissatser som utbetales pr. antall klikk. Disse prissatsene kan endres lett, og dermed kan inntektene variere og vanskelig å fastslå hvor store de blir.
- Industrien som driver slik mellomvirksomhet mellom annonsører og websidene kan bli utsatt for juks og misbruk. Slike hendelser påvirker selvfølgelig industrien og alle andre parter som er knyttet til den. Eksempel på dette er *click-fraud*⁴⁰ der annonselinkene på sidene blir ”klikket” på av automatisk programvare som genererer feil trafikk og feil utbetalinger.

4.4 Abonnentinntekter

Mange Web 2.0 bedrifter innser tidlig begrensninger som annonseinntektene gir, så de i tillegg krever en abonnentpris for å kunne bruke produkter og tjenester de tilbyr. Denne inntektsmetoden har en rekke forskjellige fordeler:

- Inntektene på denne måten er mer forutsigbare enn annonseinntekter.
- Forskjellige prissatser kan tilbys, alt ettersom hvilke funksjoner og graden av tilgang brukerne ønsker.
- Det åpner mulighet for å etablere en kundedatabase, siden alle brukere som betaler for seg er kjent. Dette kan i sin tur åpne for nye muligheter.

En av utfordringene som oppstår når bedriften krever betaling for abonnement og brukertilgang er den dramatiske nedgangen av folk som er villig til å betale for å bruke tjenester eller produkter bedriften tilbyr. Dette er hovergrunnen for at en hel rekke bedrifter velger å ikke kreve fra sine brukere. Mange brukere av tjenester og produkter dette er snakk om, forventer at utviklingen, driften, etc. betales fra annonse- eller andre inntekter. De er generelt mot å betale for å bruke tjenester og produkter av denne typen. Dette problemet kommer tydelig fram når det gjelder produkter og tjenester der alternative løsninger eksisterer. Eksempel på dette er elektronisk post, blogging, online kalendere, bildedeling, etc.

4.5 Behov for utvidelse og dominans

De fleste Web 2.0 bedrifter baserer sin virksomhet rundt det å ha stor og lojal gruppe av brukere. Dette er diskutert tidligere i rapporten. Stor og lojal gruppe av brukere utgjør en indre essens i deres forretningsmodeller. Meningen er at verdien av produkter eller tjenester skal øke for alle brukere ettersom nye brukere føyer seg gruppen av de eksisterende. Dette er en klassisk måte å utvide nettverk på, som har gjort eBay til en multimillion dollar bedrift. På den andre siden, dette store behovet for å ha brukerne skaper utfordringer. Hvis vi ser på en gruppe konkurrenter innen en it-kategori, som det gjerne er, ser vi at *alle* de kjemper for å få flesteparten av de potensielle brukerne. Denne kampen om brukerne fører til store innovasjoner som skal friste brukerne til å velge akkurat dem, men det kan også føre til at bedriftene prøver å kapre brukerne ved hjelp av andre argumenter enn det å øke verdien for andre og seg selv gjennom felleskap.

Dette fenomenet har allerede begynt å dukke opp ettersom investorer har startet å legge penger i aksjer som tilhører Web 2.0 bedrifter. Det er trolig at disse bedrifter som ikke krever stor kapital utvikling av produkter og tjenester, vil i stede bruke disse investeringer for alt de er verdt slik at mer trafikk og brukere genereres. Om ikke dette er nok, så kan bedriftene begynne å bruke useriøse metoder for å "skape" de nye brukerne slik at investorer legger inn enda mer penger i bedriftene deres. Dette inngår i en rekke faktorer som gjorde at *Internett-boblen* sprakk på 90-tallet. Når bedrifter evalueres gjennom antall klikk, brukere, sidevisninger, når det skal konkluderes om bedriftens finansielle kraft, er det alltid fare for fordreide verdier og ikke-bærekraftige vekststrategier.

4.6 Long Tail Marked

Et felles tema for mange Web 2.0 bedrifter er *long tail market*. Dette betyr egentlig at mye av det potensielle merkede består av en hel gruppe små markeder, og at sammenslåing av disse små markeder vil by på store muligheter. Dette resulterer i heller enn å fokusere på lansere mega trender, som i seg selv kan være fulle av muligheter, så fokuserer Web 2.0 bedrifter å finne måter som inkluderer millioner brukere som bruker små nisjetjenester og som ennå ikke er blitt tatt ut potensialet fra.



Figur 6, En illustrasjon av *Long Tail*^{d1} konseptet

Det er lett å komme over diskusjoner der eksempler som Amazon.com forekommer i stort antall. Det er fortsatt uklart hvordan Web 2.0 bedrifter som spiller på kreativitet skal oppnå *long tail* effekter. De ønsker å skaffe seg evner til å tjene unike behov til brukere som stadig er mer og mer kresne. Dette kan være en utfordring ikke alle mestrer. Den mest vanlige tilnærmingen et long tail marked for Web 2.0 bedrifter er å skape omgivelser for brukergenerert innhold. Strategien involverer brukere som bidrar med deres egen innhold, som i teorien er relevant til de små nisjene og spesielle interesser brukerne egentlig stammer fra.

4.7 Exitstrategier og taksering

Siden et stort antall Web 2.0 bedrifter har ikke konkrete planer hvordan de skal etablere et solid inntakts strøm og fortjeneste, er det ikke overraskende at mange av dem planlegger å bli kjøpt opp av etablerte internettaktører. Ideen er å utvikle raskt en webtjeneste, la den bli en gratis tjeneste, danne seg et stort antall aktive brukere, så selge hele konseptet til store bedrifter som Google eller Yahoo!. Fokus ligger på å danne seg stort antall brukere, i stede for å danne noe som fra dag én genererer profitt. Med dette som utgangspunkt, så vil få av slike bedrifter oppnå en bærekraftig drift. Dette lager en "race to the exit" mentalitet der de begrensede økonomiske midlene som firmaene disponerer i oppstartsfasen brukes til å lansere produkter og lage teknologier, dele dem med stort antall brukere så det blir attraktivt for de store aktører å kjøpe opp det hele. Denne modellen har fungert for et antall Web 2.0 bedrifter som *Flickr*, *upcomming.org* og nå nylig *del.icio.us*. Det ser ut som at en slik strategi er oppskriften for suksess for mange Web 2.0 bedrifter som er under oppstart, og de vil gjerne følge de samme sporene som de som har lyktes har gått. Det er et begrenset antall aktører som kan være potensielle kjøpere og som skal utvikle teknologien videre, så det er stor fare for at en hel masse bedrifter som ikke opplever en gylden tid.

Takseringen av Web 2.0 bedrifter er vanskelig og krever objektivitet. Siden det ikke finnes tradisjonelle målestokk og data som skal brukes under takseringen av slike bedrifter, er jobben vanskelig og det må vises skjønn. Takseringen foregår uten at bedriften kan vise til inntjening, skatt, fast kundemasse i tradisjonell forstand, omsetning, etc. Prisen blir gjerne bestemt av interessen kjøperen har for bedriften. Det vil ta gjerne flere år før det foreligger formelle metoder for taksering av Web 2.0 bedrifter. Metoder for taksering av de tradisjonelle⁴² bedriftene er utarbeidet for lenge siden.

4.8 Web 2.0 vs. Bubble 2.0

Kritikerne sammenlikner Web 2.0 fenomen med internettboblen på 90-tallet. *Peter Rip (Managing Director of Leapfrog Ventures)*, oppsummerer dette fenomenet på sin måte. Han presenterte det han kaller ”ny Web 2.0 likning”:

- + Mange talentfulle utviklere er klar for å starte egne firmaer
 - + Billig infrastruktur og lett tilgjengelig teknologi, samt mye tid som utviklere har til rådighet
 - + Rolige finansielle kriterier
 - + Delt kunnskap om de store tekniske trendene
 - + Relativt vanskelig å vite hva andre driver med
-
- = En overflod av bedrifter i oppstartsfasen invaderer et og samme marked

Summen av de overnevnte faktorer er at stort antall firmaer konkurrerer med hverandre og jakter på den kapitalen som er nødvendig for deres eksistens. Alle er sikre på seg selv og konkurrerer med håp om at de som *holder ut lenge nok* vil klare seg i framtiden og tjene penger. De seneste investorinteressene som er vist for Web 2.0 bedrifter styrker dette. Konkurransen er hard og det er bare en brøkdel av bedrifter – både av dem som er under oppstart og av dem som allerede er etablert – vil stabilisere seg og bli kjøpt opp.

4.9 Web 2.0 forretningsutfordringer

Hittil i oppgaven har det blitt diskutert fram en del utfordringer som Web 2.0 bedrifter står ovenfor. Her er de samlet og redefinert:

- Definisjonen av Web 2.0 bedrifter er fortsatt uklar for investorer og kunder. Bedriftene som betegner seg med *Web 2.0* skaper like mye sikkerhet som usikkerhet rundt seg selv.
- Inntjeningsmodeller og veier til profitt er ikke godt utarbeidet. For mange Web 2.0 bedrifter, spesielt i oppstartsfasen, fokuserer på å bli kjøpt opp med den ideen og teknologien de utvikler, i stede for å tenke på hvordan de kan tjene penger uten at en storebror kommer inn i bildet.
- Mange av de underliggende teknologiene som Web 2.0 bedrifter baserer seg på er ennå ikke standardisert og utbredt. Eksempel på dette er AJAX som fortsatt skaper problemer med kompatibilitet, der dette kan ødelegge for brukernes forventninger.
- Utviklingskostnader er blitt så lave at det skaper nærmest ingen hindringer i å starte en ny virksomhet. Dette tiltrekker et stort antall nye aktører som kommer på banen, men bare en brøkdel av dem vil oppnå suksess og langvarig drift.

5 Casestudier

I dette kapitlet vil flere allerede nevnte Web 2.0 bedrifter bli analysert ytterligere. Deres måter å bruke teknologi på, samt beskrivelse av forretningsmodeller de baserer seg på. Ut fra dette vil det bli lettere å trekke noen konklusjoner om potensialet og mulighetene for å etablere nye suksessfulle bedrifter i Web 2.0 verdenen.

5.1 Wikipedia



Wikipedia har blitt utviklet flere år før *Web 2.0 begrepet* kom på banen. I dag er det lett å forstå at Wikipedia er #1 når det snakkes om gode Web 2.0 applikasjoner. Hele ideen startet ut fra det å utvikle en konkurrent til tradisjonelle kunnskapsleksikoner. Daglig leder, Jimmy Walas, ansatte noen PhD utdannede folk som fikk oppgave til å skrive artikler og bygge opp en webbasert kunnskapsleksikon fra bunnen av. Det viste seg etter et år at produksjonen av artikler gikk saktere enn det som forventet. De satte i gang et lite prosjekt om å åpne muligheter for at hvem som helt kunne legge inn sine artikler gjennom et enkelt webbasert grensesnitt. Mulighetene for å endre sine og andres artikler ble også tilgjengelig etter hvert. Pr. i dag har Wikipedia tilnærmet en million artikler, og det er mer enn alle andre til sammen.

Når man ser objektivt på Wikipedia kan man se at kvalitet sikring av informasjonen ikke er på sitt beste alltid. Personlige fordommer kan framkomme i gjennom en del artikler som er lagt inn i Wikipedia. Eksempel på dette er artikkelen som omhandler Bill Gates⁴³. Artikkelen har i lenge tid vært framstilt Bill Gates på en negativ måte, og jobben han gjør er blitt kritisert. Denne og liknende artikler har siden da blitt omskrevet og mekanismer for å hindre uautoriserte brukere er blitt implementert. Motorikken som driver Wikipedia gjør det enkelt å opprette nye artikler. Disse kan ha variert kvalitet, men det er blitt enkelt å rydde opp i dem og forbedre dem.

5.1.1 Teknologi

Ut ifra navnet ser man at Wikipedia baserer seg på en wiki-teknologi⁴⁴. En wiki er en webapplikasjon som gjør det mulig å opprette og endre artikler direkte gjennom en nettleser. Applikasjonen bruker forholdsvis enkel *markup* språk. Applikasjonen er beregnet til å understøtte gruppesamarbeid og deltakelse. Dette er generelt et nøkkelfundament i Web 2.0. Wikipedia har en *radikal tillitt* til sine brukere. De mangfoldige brukerne er gull verdt for Wikipedia, men useriøse brukere kan være en belastning for Wikipedia. Wikipedia videreutvikles raskt og er applikasjon som blir bedre og bedre ettersom mange flere og flere bruker den. Desto flere det er om å skrive nye artikler og endre de eksisterende, reduserer det feil og mangler som andre måtte forårsake.

5.1.2 Forretningsmodell

Wikipedia stiftelsen⁴⁵ definerer seg på ikke-profit stiftelse som eier Wikipedia. Wikipedia innhold er tilgjengelig fritt til alle under GNU Free Documentation License⁴⁶.

Wikipedia stiftelsen eier flere andre wikisider. Dette inkludere Wikitionary, Wikibooks og Wikiquote. Stiftelsen aksepterer donasjoner for å betale for infrastruktur og de grunnleggende utgifter for driften av applikasjonen. Forslag om å finansiere Wikipedia i framtiden gjennom annonser har blitt diskutert, men disse er ennå ikke blitt realisert i praksis.

5.1.3 Wikipedia i dag

Wikipedia har blitt en av de mest besøkte websidene som er på Internett i dag. Wikipedia leverer verdifull tjeneste for brukerne. Fra det tekniske ståstedet, Wikipedia består av en hel rekke mekanismer som er definert under Web 2.0. Til tross for at Wikipedia drives av en ikke-profitt stiftelse har den blitt veldig utbredt og avansert, men samtidig enkel for å bruke. Hvis vi tenker oss at hele konsepter sitter i hendene på en bedrift som generer profitt ut av den, så ville den blitt meget suksessrik hvis den hadde introdusert mulighetene for annonsering.

5.2 Google Maps



Dette var den første applikasjonen som brukerne opplevde annerledes og som satte Web 2.0 virkelig på dagsorden. Google kom inn på et nytt allerede mettet marked med aktører og deres løsninger. Først kom de inn i søkemarkedet, og denne gangen kom de inn på markedet av webbaserte kart og revolusjonerte det hele. Google Maps er en interaktiv webbasert karttjeneste som lar brukerne kombinere sin informasjon, sette den på kartet og dele med alle andre. På lik linje med Wikipedia, så har Google Maps blitt et symbol på Web 2.0 applikasjoner og mulighetene de gir. Ved å bl.a. implementere Google Maps i en eksisterende applikasjon, så får applikasjonen innpass i Web 2.0 verden. Dette er forholdsvis enkelt på grunn av åpen API. Dette er blitt beskrevet tidligere i oppgaven.

5.2.1 Teknologi

Google Maps består av teknologier som defineres under Web 2.0. Den første, og kanskje den mest omdiskuterte, er AJAX. Bruken av AJAX teknologien gjør at de visuelle opplevelsene av applikasjonen blir virkelig i en sær klasse i forhold til konkurrerende løsninger. Den andre aspekten ved Google Maps som gjør applikasjonen Web 2.0 vennlig er mulighet for å bruke applikasjonen som en modul til en annen applikasjon. Applikasjonen støtter dette enkel gjennom sin API. Interessen for Google Maps ble fort stor, så ble koden for denne tjenesten fort knekt og de første applikasjonene som baserte seg på Google Maps begynte å dukke opp. Eksempel på denne er Housing Maps⁴⁷. Raskt tid etter dette ble Google Maps API gjort fritt tilgjengelig for alle og en hel ny generasjon av applikasjoner basert på Google Maps begynte å eksistere på nettet.

5.2.2 Forretningsmodell

På lik linje med alle de andre applikasjonene Google utvikler, så har Google Maps sin API fritt tilgjengelig for alle. I tillegg til brukervennligheten, så har dette vært en viktig årsak til at denne tjenesten er blitt fort akseptert., I motsetning til Wikipedia, Google er en profittpasert bedrift som tjener penger gjennom annonse andre betaler dem å ha på Google sine tjenester. Alle som er interessert kan kjøpe annonseplass på profilerte steder, f. eks blant søkeresultater av Google Maps. Mange andre applikasjoner og websider som knyttes til Google sine tjenester lar Google annonse dukke opp deres egne sider og applikasjoner. Dette generer penger til Google og dem selv. I tillegg vises "Powered by Google" logo på hvert kart Google Maps viser, og dette styrker Google sin brand.

5.2.3 Google Maps i dag

Google Maps har satt standard for hvordan Web 2.0 applikasjoner skal bygges og hvordan pengene kan tjenes ut av dem. I stedet for å lage applikasjon som folk betaler for, så laget Google applikasjon med mange Web 2.0 funksjonaliteter som brukerne liker å bruke og som de bruke mye. Dette åpnet mulighet for store annonseinntekter, samtidig som Google brand blir verdens kjent og aksjeverdier stiger.

5.3 del.icio.us



del.icio.us er en sosial bokmerkingsside som er nevnt tidligere i oppgaven. Denne tjenesten tillater brukere å dele deres bokmerker (bookmarks), kommentere dem og sette tagger på dem. Tags brukes som universelle kategorier, og det er effektivt å søke gjennom disse kategorier. I stede for å søke med Google for nyttige websider, så kan man gjøre søk gjennom bokmerker som er lagt inn i *del.icio.us* av andre brukere som har testet ut disse linkene på forhånd og synes godt om dem. Denne tjenesten åpner mulighet for å ha direkte tilgang til sine egne bokmerker, uavhengig av hvor man er i verden, hvilken datamaskin man sitter ved eller hvilken nettleser man bruker.

5.3.1 Teknologi

Den primære Web 2.0 funksjonaliteten som *del.icio.us* bruker er tagging. Tags tillater brukerne å filtrere ut bokmerker og knytte dem opp mot andre, slik at navigering mellom websider som er i samme kategori gjøres naturlig.

5.3.2 Forretningsmodell

del.icio.us er tjeneste for fri bruk og krever ikke noe abonnentbasert brukertilgang. Det er heller ikke noe annonse som vises på websiden. Josh Schachter er gründer av denne tjenesten, og det er kjent at han har mottatt finansielle tilskudd fra flere bedrifter, organisasjoner og enkelt individer. Noen av disse er: Amazon.com, Marc Andreessen, Esther Dyson, Seth Goldstein, Tim O'Reilly, and Bob Young. Schachter og hans investorer har utalt de ennå ikke er sikre hvilken forretningsmodell denne tjenesten skal basere seg på. Pengene som er allerede investert, brukes til å utvide funksjonalitet til tjenesten og forbedre brukervennlighet. Det kan tenkes at målrettet annonse tas i bruk for å skape en inntaktskilde. På den andre siden ville det ikke være urimelig å krave betalt for brukertilgang som gir utvidet funksjonalitet.

5.3.3 del.icio.us i dag

del.icio.us er også en Web 2.0 som brukes av mange, men den mangler god forretningsplan på lik linje med de fleste andre Web 2.0 applikasjoner. Det spekuleres at *del.icio.us* vekker interesser for det meste hos brukere som er teknisk anlagt. De mest populære tags det søkes om er "programming," "software," "ajax," og "linux". På den andre siden satte disse spekulasjoner ingen hindring for at Yahoo! kjøpte *del.icio.us* for ca. et år siden. Denne nyheten ga nytt liv i alle forhåpninger om at Web 2.0 bedrifter som ikke har en utarbeidet forretningsplan eller klare formeninger om hvordan de skal tjene penger, er attraktive for oppkjøp av de store aktørene på markedet.

5.4 flickr



The best way to **store, search, sort and share** your photos.

Flickr er kanskje det beste eksemplet på en applikasjon som er utviklet av en bedrift mens den fortsatt i er oppstartsfasen. Flickr tar også effektivt i bruk tilgjengelige Web 2.0 teknologier, samtidig som flickr baserer seg på en gjennomførbar forretningsmodell. Flickr kom på markedet mens det var flere alternative løsninger for bildedeling over Internett. De fleste løsninger hadde en begrenset adgang til bruken av applikasjonen, samt deling av bildene. Flickr tok en ny vending og kom opp med flere mulige løsninger på hvordan brukerne kunne få tilgang til sine egne og andre sine bilder. I tillegg la de det opp på en slik måte at det nærmest oppleves om å leke med bildene. De tok i bruk innovative måter å gruppere bilder på, matematiske algoritmer som hjelper brukerne å finne fram til det som interesserer dem og på en effektiv måte navigere gjennom bilder og søkeresultater.

5.4.1 Teknologi

På lik linje med Google Maps, så er flickr et symbol på en Web 2.0 applikasjon. En rekke funksjonaliteter som flickr tar i bruk bekrefter dette. Flickr tar i bruk *folksonomy*, dette gjør det mulig for brukeren å sette tags på sine bilder som effektiviserer søking etter bilder for andre de brukerne. Med dette finner de effektivt til en stor variasjon av bilder fra like stor variasjon av brukere som har lagt dem inn. Flickr bruker *popularitets rangeringer* i for av *favoritter* og *mest interessant*. Dette øker aktiviteten hos brukerne som legger inn flere og flere bilder som i sin tur skaper økende nettverkseffekter. Flickr tar også i bruk skreddersydde RSS feeds. Denne teknologien lokker flere brukere og vekker interessen for flickr. Det ligger også tilgjengelige API som gjør det mulig for utviklere av andre websider å flette inn bilder fra flickr i sine blogger f. eks.

5.4.2 Forretningsmodell

Flickr bruker flere forskjellige inntjeningsstrategier for å generere inntekt. *Målrettet annonse*⁴⁸ er en strategi. En annen strategi er betaling av abonnement. Dette gir brukeren en utvidet funksjonalitet. Bl.a. så betyr dette ubegrenset lagringsplass og datatrafikk. Abonnement koster 25 Dollar for et helt år. Framkalling av bilder er også en tjeneste flickr yter overfor sine brukere. Det er kanskje riktig å omdefinere brukere til fullkommende kunder når det gjelder flickr. I dette eksemplet er det allerede snakk om kombinasjon av elektroniske og fysiske tjenester de gjør for sine brukere/kunder. Flickr ble kjøpt opp av Yahoo! for om lag 25 millioner Dollar i 2005.

5.4.3 flickr i dag

I utgangspunktet var ikke flickr en forretning i tradisjonell forstand. Det var heller bare en applikasjon med en forretningsmodell og definerte inntjeningsstrategier. Slike egenskaper kunne hvilken som helst annen Web 2.0 bedrift ha, men flickr stoppet ikke med det. I stede for å stole utelukkende på annonseinntekter, flickr utviklet alternative måter å tjene penger på. Det ikke lett å finne ut hvor mye penger de tjente gjennom disse strategier, men flickr har skapt verdier slik at bedriften kom i en attraktiv posisjon og levde ut en mer lønnsom exitstrategi enn de fleste andre Web 2.0 bedrifter.

6 Metoder

Dette kapitlet vil omhandle alt det jeg har gjort for å få det resultatet vi har i dag. Jeg skal prøve å komme med en grundig beskrivelse for å få fram både hva som er laget og hvorfor det er. Det skal være mulig for leseren å kunne kopiere framstilt implementasjon og få samme resultat i et forsøk. All beskrivelse er forsøk å være ”ovenfra og ned”. Det betyr først at designet er presentert på et oversiktlig blokknivå slik at leseren får en idé om hva som er hovedstrukturene i det som er laget. Deretter går jeg inn på detaljer i en passende rekkefølge

6.1 Kort om informasjonssystemer

Informasjonssystemer finnes overalt i det moderne samfunn og i en rekke ulike versjoner. Generelt kan et IS defineres som et system for innsamling, bearbeiding, lagring, overføring og presentasjon av informasjon. Det finnes både manuelle og databaserte systemer. Databaserte informasjonssystemer er spesielt egnet der det kreves raskere bearbeiding av data, lagring og gjenfinning av store mengder data, større nøyaktighet i prosesser / resultater og der manuelle inngrep i verdikjeder må reduseres. Det finnes en rekke ulike informasjonssystemer samt en rekke forskjellige klassifiseringer av disse. Informasjonssystemet som knyttes til denne oppgaven faller under klassen transaksjonsbaserte systemer. Dette er online systemer som støtter den daglige drift av virksomheter og kjennetegnes ved et høyt antall transaksjoner som består av relativt enkle funksjoner.

6.1.1 IS utvikling

IS utvikling eller systemutvikling omfatter alle aktiviteter fra problemdefinering til et system er ferdig implementert og satt i drift. Et system kan befinne seg i ulike tilstander avhengig av hvor i utviklingsprosessen en er. Dette blir ofte referert til som livssyklusen til et informasjonssystem der hver tilstand er resultatet av en rekke aktiviteter som inngår i systemutviklingen:

- **Problembeskrivelse:** fokuserer på virksomhetens muligheter og problemer og resulterer i en problembeskrivelse som klart definerer målsetning, omfang, etc.
- **Analyse:** deles ofte inn i virksomhetsanalyse og systemanalyse. I første del analyseres virksomheten for å avgjøre hvordan systemet kan støtte den. Deretter vurderes og bestemmes systemets innhold. Resulterer i en kravspesifikasjon.
- **Design:** hvor en først utvikler en prinsipiell teknisk løsning og deretter utvikler en prinsipiell utstyrstilpasset teknisk løsning
- **Implementasjon:** realisering av systemet inkludert testing og det som behøves for å sette systemet i drift. Resulterer i et system klar for å ta seg av oppgaver der systemet overleveres til dagligdrift og vedlikehold
- **Drift og vedlikehold:** sørge for at systemet er tilgjengelig for brukerne, foreta feilretting når feilsituasjoner oppstår og små forbedringer av systemet. Hvis større forbedringer må foretas, blir dette ofte betraktet som et utviklingsprosjekt hvor alle fasene blir gjennomgått

6.1.2 Uviklingsmodeller

Det finnes en rekke forskjellige utviklingsmodeller og de kan grovt grupperes i disse hovedgrupper:

- **Kod og fiks modellen:** utvikleren ble presentert et problem og satte umiddelbart i gang med å utvikle programmer for deretter å rette / korrigere koden når feil ble oppdaget. Krav, konstruksjon, etc. ble håndtert underveis i programmeringen.
- **Fossefall/vannfallsmodellen:** "waterfall" er en type stegvis modelle der feedback løkker mellom nabofaser tillates og prototyputvikling går i parallell med kravspesifikasjon. Vannfallsmodellen er den mest utbredte og en rekke forskjellige varianter av denne har blitt utviklet.
- **Evolusjonsmodellen:** inndeler utviklingen i et sett av steg der løsningen gradvis blir utvidet basert på erfaringer i hver fase. Iterasjon og prototyputvikling er sentrale begreper i denne modellen. I iterativ systemutvikling gjentas aktiviteter der omfanget av løsningen gradvis økes for hver iterasjon. Ved prototyputvikling utvikles en prototyp på et tidlig tidspunkt i prosessen og videreutvikles og kan eventuelt resultere i det systemet.
- **Spiralmodellen:** fokuserer på risiko og reduksjon av risiko i systemutviklingsprosessen. Utvikling følger en spiralmodell og deles i et sett av syklus i spiralen. Hver syklus starter med definering av målsetning, alternative måter å realisere løsning for dette og identifisering av begrensningene. Deretter følger evaluering av løsningsalternativer inkludert identifisering av risiko, forslag til reduksjon / eliminering av risiko, plan og aksept for neste syklus. Risiko håndteres systematisk fra syklus til syklus i spiralen

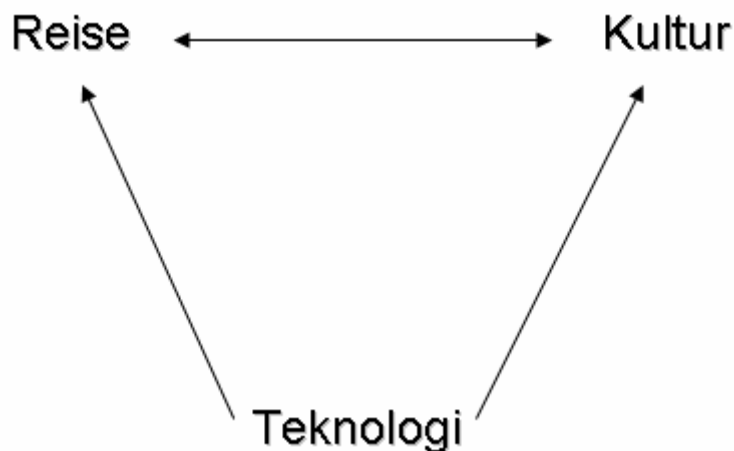
6.1.3 Modelleringspråk

Det finnes mange klassifiseringer av modelleringspråk. De kan grupperes etter hvilket perspektiv (orientering / modelleringsretning) de støtter, hvor perspektiv blir bestemt av språkets kjernebegreper og prinsipper. Eksempler er dataorienterte språk som har fokus på hvilke data som behøves f.eks. ER. Prosessorienterte språk som har fokus på hvordan systemet skal fungere/hva det skal gjøre f.eks. dataflytdiagram. Objektorienterte språk som har fokus på objekter og meldingsutveksling mellom disse f.eks. UML og OOA

ER – språket som senere i oppgaven skal beskrive systemet som jeg har laget, baseres seg på entiteter, relasjoner (forbindelser) og attributter.

6.2 Autentisk reising

Autentisk reising er fenomenet som beskriver en reisemåte, samt måten å organisere reisefasiliteter og kulturer på. Med andre ord betyr dette hva de reisende ønsker å oppleve, samt fasiliteter tilbydere har for de reisende. Tilbydere som assosierer seg med fenomenet, fokuserer på å gi de reisende opplevelser som er autentiske for det bestemte reisemålet. Autentisk reising er representert med 2-3 portaler på nettet. Pr. dags dato vil disse aktører ikke konkurrere med informasjonssystemet som baserer seg på studien i forbindelsen med min besvarelse. Jeg setter fokus på å realisere et prosjekt som skal åpne mulighet for informasjonsutvikling gjennom deltakelse. Dagens trender beskrevet i Casestudier - kapitlet tilsier at brukerne skaper gjerne informasjonen ved å dele sine opplevelser og meninger med andre via nettet.



Figur 7, Reise – Kultur – Teknologi

Modellen over viser oss interaksjonen mellom begrepene *reise*, *kultur* og *teknologi*. Denne modellen ligger i bunnen for det systemet jeg har utviklet. Poenget var å ta teknologien i bruk på en slik måte at den støttet oppunder relasjonen mellom reise og kultur.

6.3 Prosjektoversikt

En klar prosjektoversikt gir etter min mening et klarere bilde av prosjektmål, programvarebeskrivelse, hovdeaktiviteter, delmål og ressurser som var tilgjengelig i dette prosjektet.

6.3.1 Prosjektmål

Dette er hovedmål med utviklingen av webapplikasjonen som skal tjene organisasjonen Authentic Travel:

- Gi mulighet for variert interaktivitet
- Gi mulighet for sosialt nettverk og effekter det skaper
- Gi organisasjonen og brukerne verktøy for utvikling og drift av informasjonssystemet
- Informasjonsutvikling gjennom deltakelse

Et sett av moduler og funksjonaliteter de består av skal gi organisasjonen og brukeren *mulighet for variert interaktivitet*. Dette skal muliggjøre et naturlig samlingspunkt for alle som deler interessen for autentisk reisemåte og autentiske reismål. Dette gjelder de reisende og de aktørene som tilbyr slike opplevelser.

Dette vil *gi mulighet for sosialnettverk og effekter det skaper*. Med dette menes nettverk som skal kunne dannes mellom folk som deler samme interesse, som i sin tur skal kunne skape relasjoner folk imellom som de setter pris på.

Authentic Travel informasjonssystemet vi *driftes av brukerne* gjennom informasjonene de legger inn, *samt administratorene* av applikasjonen som skal holde kontrollere at innlagt informasjon ikke virker støtende på andre. Hele webapplikasjonen skal være webbasert og kunne brukes fra alle datamaskiner med en nyere nettleser og internett tilgang.

Denne interaksjonen mellom aktørene og deres deltakelse skal være grunnmotorikken i *utviklingen og driften av informasjonssystemet*. Informasjonen som legges inn vil være verdi skapende.

I sin helhet vil all tilgjengelig funksjonalitet være med på å skape et miljø for diskusjon, nyttig informasjon, søking og visning av den gjennom de interne tjenestene, samt gjennom par eksterne tjenester. Informasjonen som systemet skal basere seg på skal kunne manipuleres på mange forskjellige måter.

6.3.2 Kravspesifikasjon

Kravspesifikasjonen som det nye systemet skal basere seg på er ikke fastlagt på forhånd. Kravene til systemet har mer eller mindre framkommet gjennom tiden under utviklingsprosessen. Organisasjonen Authentic Travel har pr. i dag ingen eksisterende liknende system, så det nye systemet sees ikke på som en oppdatering av noe som allerede finnes til en viss grad. Vi ville skape et system som baserte seg på Web 2.0 eksisterende teknologier, samt komme opp med ny funksjonalitet og teknologi som gjør det hele unikt innen sitt område. Kravspesifikasjon var aldri definert i tradisjonell forstand, men det hele ble en utviklingsprosess som hele tiden resulterte i noe nytt.

Hovedmotivet bak utviklingen var et overordnet krav og det er å skape en portal der *tilbud og etterspørsel* møtes.

6.3.3 Delmål

Prosjektet baserer seg på fem delmål. Det er blitt satt faste datoer for å oppnå delmålene, hva som er forventet omfang av det som skal oppnås. Milepæler er et styringsinstrument jeg brukte for å visualisere realiseringen av delmålene. Framgang og videre arbeid med gjenstående delmål var avhengig av godkjenning av foregående fra veilederens side. Første delmål i forbindelsen med prosjektet måtte godkjennes av IFI - administrasjonen.

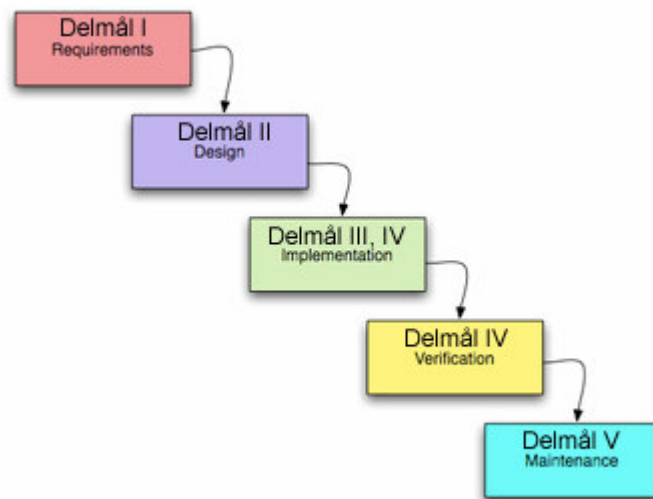
Aktivitet	Beskrivelse
Delmål I	Skrive essay relevant til oppgaven. Definere oppgaven.
Delmål II	Use Case Modell ER diagram for databasestrukturen
Delmål III	Implementering av allerede eksisterende moduler: forum og blogg
Delmål IV	Design av ny funksjonalitet Programmering og implementering av ny funksjonalitet, samt kontroll
Delmål V	Testing av applikasjonen ved hjelp av testbrukere, samt JMeter ⁴⁹ Vedlikehold

Tabell 2, Delmål for prosjektet

6.3.4 Prosessmodell

I dette prosjektet kunne kravspesifikasjonen bli ganske omfattende hvis vi skulle dekke alle systemets funksjonaliteter. På ett tidspunkt forutsatte vi at graden av spesifikasjonen vi definerte hittil er endelig, og at det ikke vil komme store endringer av kravspesifikasjonen senere i systemutviklingsprosessen. Systemet er i en BETA fase, og det vil si at det er under videre utvikling. Det framstår i dag med de manglene som er.

Ut ifra disse forutsetningene baserer min prosessmodell på den klassiske Royce vannfallsmodellen⁵⁰.



Figur 8, Vannfallsmodell – prosessene strømmer etter hverandre

Etter vannfallsmodellen vil hver fase gjennomgås fullstendig før man beveger seg til neste fase. Vannfallsmodellen passer godt for prosjekt av størrelsen jeg jobbet med i denne sammenhengen fordi det er overkommelig å fullføre en fase innenfor rimelig tid.

6.4 Risikovurdering

Dette punktet omhandler de syv mest relevante risikoene som knyttes til dette prosjektet. Disse risikoene kunne hver for seg eller kombinasjon av flere ødelegge mye for utviklingen av systemet.

6.4.1 Erfaringsmangel (R1)

Dette beskriver risikoen for at jeg ikke er erfaren i å drive med denne typen arbeid er den klart største risikoen. Dette er det største prosjektet i studierelatert sammenheng og jeg står helt alene med det. Tidligere erfaringer baserer seg stort sett på gruppearbeid.

6.4.2 Verktøyproblemer (R2)

Med bestemte tidsfrister og mye arbeid kan det fort bli et problem dersom utviklingsverktøyene ikke virker tilfredsstillende.

6.4.3 Sykdom (R3)

Dersom jeg blir plaget av sykdom som setter meg ut i lengre perioder, så vil dette medføre utsettelse og problemer.

6.4.4 Frafall (R4)

Dersom jeg ikke klarer utfordringene som oppstår når men jobber med en oppgave av slik omfang, og bestemmer meg for å trekke meg fra videre arbeid.

6.4.5 Ikke godkjente delmål (R5)

Dersom jeg ikke får godkjent mellomleveringene, dvs. ikke oppnår delmålene vil dette gi problemer med å opprettholde kvaliteten gjennom hele prosjektet.

6.4.6 Missforståelser av kravspesifikasjonen (R6)

Feiltolkning av kravspesifikasjonen kan fort føre til at jeg tar på seg mer arbeid enn jeg burde eller at arbeidet må gjøres om igjen.

6.4.7 Hardwareproblemer (R7)

Det mest alvorlige her kan medføre at alt eller store deler av arbeidet blir tapt. Dette kan skje ved diskkrasj eller at dataene på andre måter skades for eksempel gjennom et ødelagt filsystem på arbeidsstasjonen min eller på serveren der systemet blir satt opp.

Risiko	Fare for at det inntreffer	Tidspunkt
R1	Middels	Hele tiden
R2	Middels	Design, implementasjon
R3	Middels	Hele tiden
R4	Liten	Hele tiden
R5	Middels	Hele tiden
R6	Stor	I starten av prosjektet
R7	Middels	Hele tiden

Tabell 3, Risikovurdering

Risiko	Håndtering
R1	Denne risikoen er den som er vanskeligst å håndtere. Den viktigste måten å overkomme denne risikoen på er å legge til ekstra med tid i alle faser.
R2	Denne kan håndteres ved å teste verktøy i rimelig tid før de skal brukes og dermed kunne finne og utbedre de problemene som oppstår.
R3	Her finnes det få gode muligheter utenom snarest mulig kurering av sykdommen.
R4	For å håndtere denne risikoen har jeg allerede tatt opp problemstillingen med meg selv i mange år. Jeg har motivasjon nok til å stå på løpet ut.
R5	Denne risikoen kan kun motvirkes ved å komme tidlig i gang med det som skal gjøres, slik at brukbar framgang vises fram ved neste time hos veilederen
R6	Denne risikoen vil jeg bearbeide ved å være flink til å utfordre veilederen til å komme med innspill og avklaringer underveis i prosjektet.
R7	Hardwareproblemer vil forhåpentligvis ikke være et stort problem. Alt arbeid på serveren er lagret hos en seriøs leverandør som sikrer gode rutiner for backup av filer. Lagring av gjort arbeid privat hos meg vil forgå på to separate datamaskiner.

Tabell 4, Risikohåndtering

6.5 Tekniske konsekvenser

Etter mine vurderinger virker det som om det skal bli minimale tekniske konsekvenser ut ifra den kravspesifikasjonen som er blitt definert. Kravspesifikasjonen beskriver funksjonaliteten til systemet og den infrastrukturen som organisasjonen besitter nå vil være nok for at systemet skal kunne kjøre på deres egen webserver.

Systemet vil bli realisert ved hjelp av standardkomponenter i PHP⁵¹, noe som vil luke ut de fleste problemene med tanke på systemkjøring på en webserver og tilgang til systemet fra klienter

Etter å ha vurdert de forskjellige kravene som har blitt fremstilt i kravspesifikasjonen, var det naturlig å foreslå en klient/webserver arkitektur. Systemet skal kjøre på webserveren til organisasjonen, som viser seg å støtte kjøring av PHP og MySQL⁵² database. Klientapplikasjon vil være den ordinære nettleseren som alle datamaskiner har installert.

6.6 Usecase

Termen usecase er en god oversettelse av begrepet *use case* som beskriver hvem vil bruke de spesielle funksjonene i systemet for å oppnå et mål. Dette blir en overordnet spesifisering av systemets funksjonalitet. Dette er en *black box* tilnærming til beskrivelse av systemet. Usecase uttrykker ikke fullverdig en virksomhetsprosess, men viser hvor systemet blir benyttet i en slik prosess

6.6.1 Aktører

Jeg har definert følgende aktører. Aktører er brukere av systemet i en bestemt rolle. Når det er sagt, så betyr det ikke at det må absolutt være en person. En aktør kan også være et eksternt system. Aktøren er eksternt i forhold til systemet, og aktøren interagerer med systemet. Det er også riktig å si at aktøren har krav til systemet.

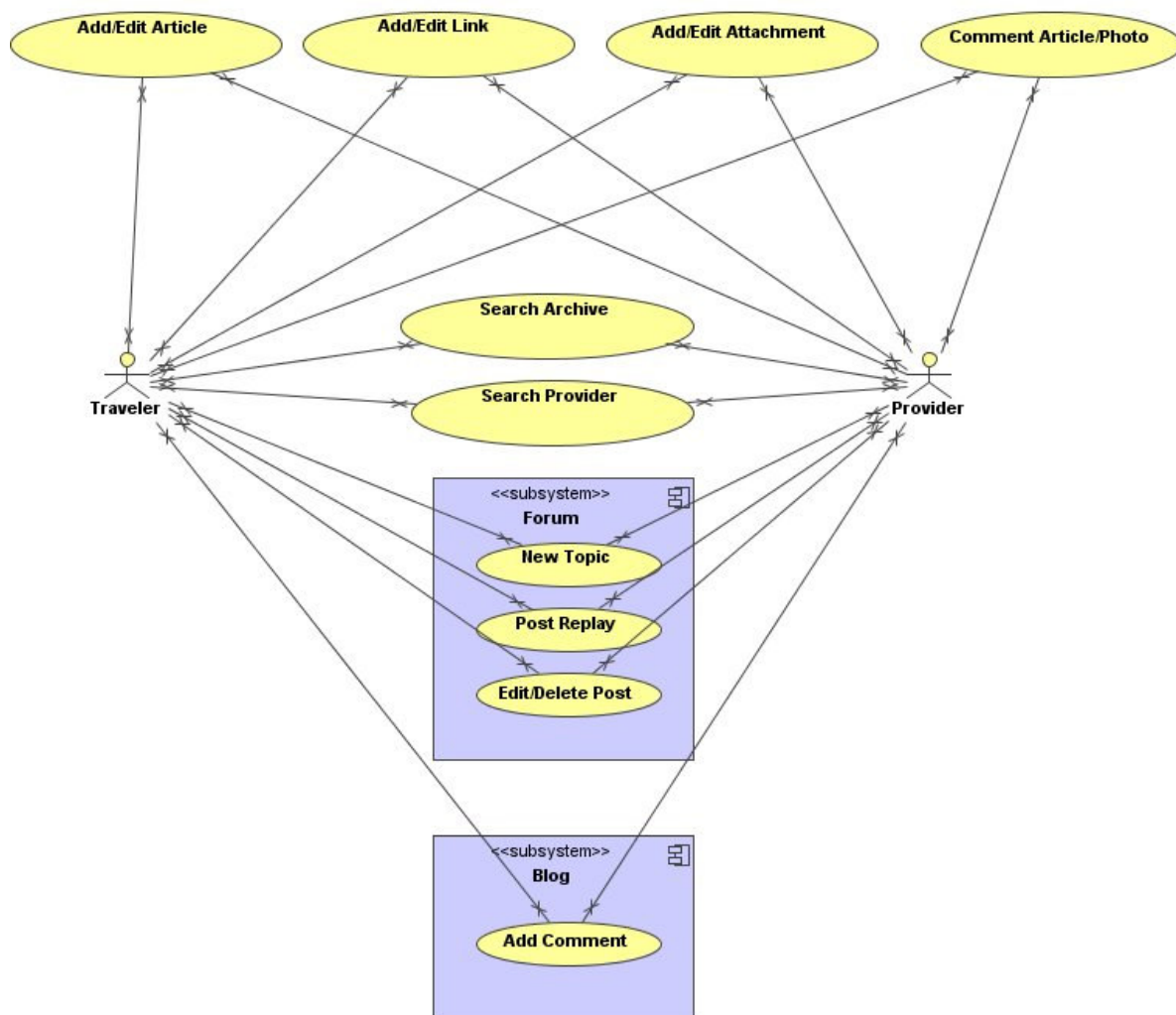
Aktør	Beskrivelse	Eksempel
Reisende (Traveler)	Person som er opptatt av reising.	Personer av denne typen bruker systemet til å legge inn egne opplevelser de har hatt mens de var på reisefot. De bruker også systemet til å finne informasjon andre har skrevet, samt planlegge de kommende turene.
Tilbyder (Provider)	Person som tilbyr opplevelser for de reisende	Personer av denne typen har gjerne en virksomhet innen turistnæringen. De bruker systemet til å profilere sine egne tjenester de yter. Den innlagte informasjonen vil hjelpe de reisende til å finne opplevelser som finnes.
Administrator (Admin)	Person som administrerer systemet	Personer av denne typen er gjerne en del av organisasjonen. De administrerer systemer, sensurerer det som trengs, manipulerer innlagt informasjon, sperrer brukerkontoer, ol.

Tabell 5, Aktører

Disse aktørene gir opphav til en rekke forskjellige usecase. Disse er beskrevet i de to kommende diagrammene. Diagrammene er delt opp etter aktører som knyttes til dem. Dette er gjort på grunn av plassen som er tilgjengelig i et slikt dokument, samt at det virker mer oversiktlig. Forskjellen mellom aktørene *Traveler* og *Provider* er ikke stor når det gjelder tilgang til systemet de har. *Provider* har bare en ekstra fasilitet, og det er å bli registrert i *Provider register*. Med basis i dette registrert, skal systemet ha en tjeneste som er tilnærmet "Gule Sider". Det kan hende at tilbyderinformasjon skal knyttes opp mot hvert innlegg noen har hatt på veiene av denne registrerte tilbyderen. Dette vil da øke verdien av innlegget og være en verdifull annonse for tilbyderen. Denne tjenesten er pr. i dag ikke implementert.

6.7 Funksjonalitet for aktører *Traveler* og *Provider*

Det er lett å se at *Traveler* og *Provider* deler de samme bruksmønstrene. Hvorfor regnes de ikke da som én og samme aktør? Jeg ser en klar skille mellom intensjonen de to forskjellige aktørene har når de tar systemet i bruk. I tillegg er dette bevist gjort hvis systemet ved senere anledning skal utvides og det er bare aktøren *Provider* som for eksempel skal bare få utvidet tilgjengelig funksjonalitet. Disse to tilsynelatende aktører vil etter min mening alltid oppføre seg annerledes mot systemet.



Figur 9, Usecase for *Traveler* og *Provider*

6.7.1 Add/Edit Article

Denne funksjonaliteten gjør det mulig for aktøren å legge inn sine egne artikler om det de måtte finne ønskelig. Innlegget vil bli registrert i databasen og samtidig kunne vises på hovedsiden, i kategorien artikkelen er blitt i, samt vises som søkeresultat dersom de tilgjengelige søketjenester benyttes.

Search Article

Title

Tags

List of Article

Total Records: 3

Title	Tags	Online
Lofoten ranked as 2nd most beautiful island in the world!	lofoten norway	Yes
WELCOME TO GUDBRANDSDALEN	Gudbrandsdalen Norway	Yes
The Maelstrom	lofoten norway fishing	Yes

[Add New](#) [First](#) [Prev](#) 1 of 1 [Next](#) [Last](#)

Add Article

Id Random

User Id

Date Of Input

Picture

Picture Title

Photographer

Country

Title

Subtitle

Text

Tags

Online ☒

Figur 10, Brukergrensesnitt – Add/Edit Article

Jeg har lagt vekt på brukervennlighet når jeg har utviklet grensesnitt brukerne forholder seg til. Gjennom dette grensesnittet er det mulig for brukeren å se de tidligere skrevne artikler, endre deg eller slette permanent. Hver artikkel har et attributt som heter *Online*. Denne gjør det mulig for å bestemme om artikkelen skal vises offentlig eller om den blir i kladdtilstand. Det finnes funksjonalitet som tillater kommentering av artikler.

6.7.2 Add/Edit Link

Denne funksjonaliteten gjør det mulig å legge ved tilleggslinker til en allerede skrevet artikkel. Linkene vil gjerne føre til relevante sider, dokumenter, og lignende. Disse linkene vil også merkelapper (tags) på seg, så det blir mulig å søke etter linker gjennom den interne søke tjenesten. Web 2.0 webapplikasjoner tar kraftig i bruk *tags* som teknologi. Det samme ville jeg også gjøre slik at dokumenter, linker, bilder med samme eller tilnærmet like tagger skal kunne slås sammen i eventuelle søkeresultater eller vanlig navigering. Denne teknologien er mest kjent gjennom tidligere beskrevet *del.icio.us*

Search External Link	
Link Title	<input type="text"/>
Url	<input type="text"/>
Description	<input type="text"/>
Tags	<input type="text"/>
<input type="button" value="Search"/>	

List of External Link	
Total Records: 6	
Link Title	
Gudbrandsdalen	
Lofoten Islands	
Gallery	
General Information	
Lofoten ranked as 2nd most beautiful island in the world!	
Lofoten ranked as 2nd most beautiful island in the world!	
First Prev 1 of 1 Next Last	

Add/Edit External Link	
User Id	at <input type="button" value="v"/>
Article Id	WELCOME TO GUDBRANDSDALEN <input type="button" value="v"/>
Country Id	Norway <input type="button" value="v"/>
Link Title	Lofoten ranked as 2nd most beautiful island in the world
Url	http://www.finnning.no/at/redirect.php?article_id_randor
Description	The Lofoten Islands in Northern Norway have been ranked by The Observer newspaper as no. 2 on a list of the world's 20 most beautiful islands.
Tags	lofoten norway
<input type="button" value="Submit"/> <input type="button" value="Delete"/>	

Figur 11, Brukergrensesnitt – Add/Edit Link

6.7.3 Add/Edit Attachment

Denne funksjonaliteten gjør det mulig for brukerne å legge ved tilleggsbilder som vedlegg til en allerede skrevet artikkel. Dette vil da danne et bildegalleri som skal supplere den tekstlige informasjonen som er lagt inn i systemet. Funksjonalitet for å kommentere de innlagte bildene er til stede. Denne teknologien er mest kjent fra tidligere beskrevet flickr.

Search Photo	
Title	<input type="text"/>
Tags	<input type="text"/>
<input type="button" value="Search"/>	

List of Photos	
Total Records: 1	
Title	Test
Add New First Prev 1 of 1 Next Last	

Add Photo	
User Id	at <input type="button" value="v"/>
Article Id	Select Value <input type="button" value="v"/>
Country Id	Select Value <input type="button" value="v"/>
Filename	<input type="text"/> <input type="button" value="Browse..."/>
Title	<input type="text"/>
Author	<input type="text"/>
Tags	<input type="text"/>
<input type="button" value="Add"/>	

Figur 12, Brukergrensesnitt – Add/Edit Photo

6.7.4 Comment Article/Photo

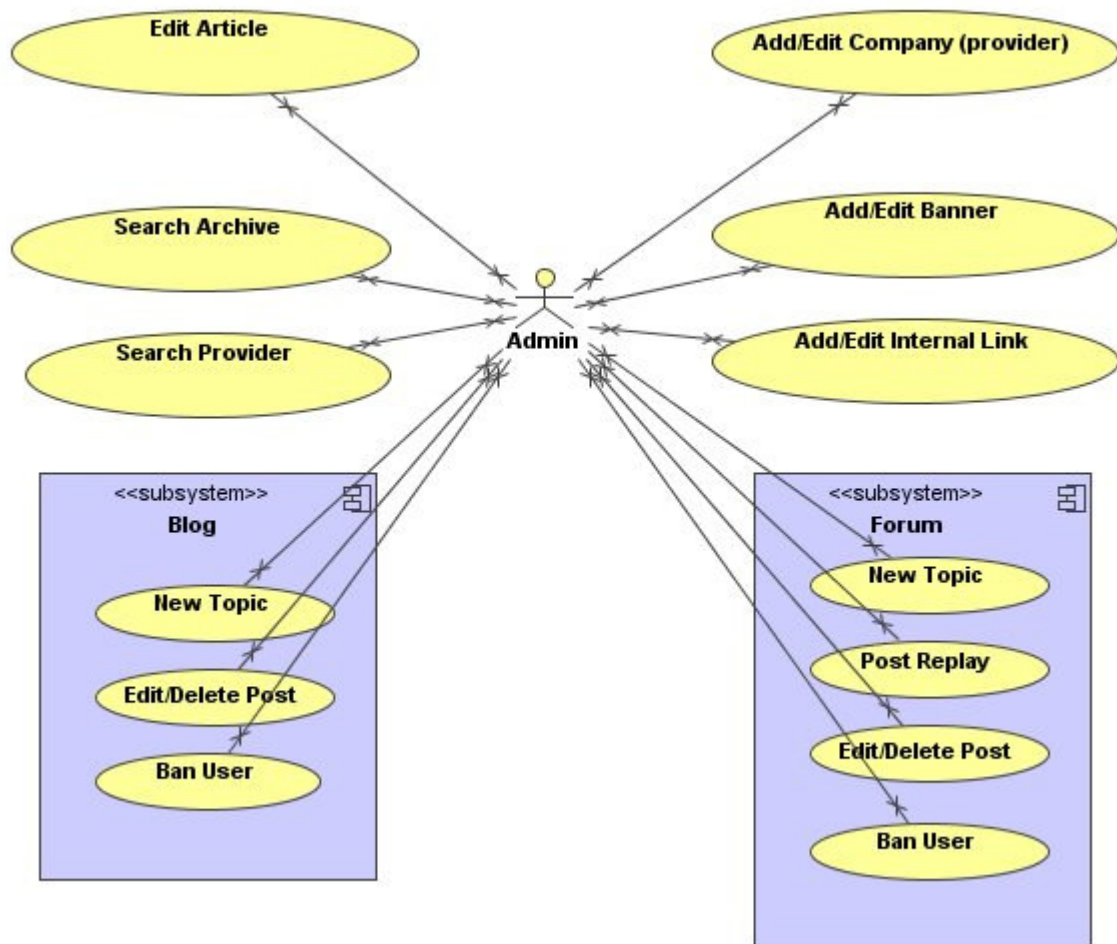
Denne funksjonaliteten gjør det mulig å kommentere artikler og bilder som er lagt inn i systemet. Selv om artikkel og bilde er to forskjellige enheter i systemet, deles det samme brukergrensesnitt for å kommentere dem.

Add/Edit Comment	
User Id	at <input type="button" value="v"/>
Date Of Input	10/23/2006 2:56:31 PM
Comment	<input type="text"/>
<input type="button" value="Add"/>	

Figur 13, Brukergrensesnitt – Add/Edit Comment

6.8 Funksjonalitet for aktøren Admin

Funksjonaliteten som aktøren Admin har er i stor grad helt forskjellig fra den de to andre aktørene har tilgang til. Vi ser at den største likhet oppstår i bruken av sub systemer Forum og Blog. Selvfølgelig er det her også en del fundamentale forskjeller mellom vanlig bruker og admin, men jeg kommer ikke inn på dette i denne omgang.



Figur 14, Usecase diagram for Admin

6.8.1 Edit Article

Denne funksjonaliteten gjør det mulig for administratoren av systemet å redigere den mest vitale informasjonen av en artikkel som er lagt inn av andre brukere. Det er menning at administrator skal kunne endre på informasjon som måtte trenge det, samt være i stand til å slette annonse permanent eller sette den *offline*.

List of Article	
Total Records: 3	
Title	Online
Lofoten ranked as 2nd most beautiful island in the world!	Yes
WELCOME TO GUDBRANDSDALEN	Yes
The Maelstrom	Yes

Figur 15, Edit Article

6.8.2 Add/Edit Company (provider)

Denne funksjonaliteten gjør det mulig for administrator å registrere tilbydere av autentiske reisemål og opplevelser. Informasjonen fra dette registrer skal brukes i modulen som kan sammenliknes med "Gule Sider". Dessuten er det påkrevd at tilbyder er registrert i databasen på denne måten hvis det blir ønskelig for tilbyderen å ha annonseannonser på Authentic Travel informasjonssystemet.

List of Company Pages	
Total Records: 2	
Company Title	
Lofoten Startside	
Lofoten Golf Links	

Figur 16, Brukergrensesnitt – Add/Edit Company (provider)

6.8.3 Add/Edit Banner

Denne funksjonaliteten gjør det mulig for administrator å legge inn bannere i systemet som da vil vises på alle sidene av informasjonssystemet. Slik det er nevnt tidligere så er dette systemet helt webbasert, dermed var det også på sin plass å lage funksjonalitet for annonseutplassering på informasjonssystemet. Alle registrerte tilbydere har mulighet til å annonsere. Dette vil bli en betaletjeneste.

Search Advertisement

Title:
Company Title:

List of Advertisements

Total Records: 4

Title	Company Title	Online
Banner for november	Lofoten Startside	Yes
Banner for year 2006	Lofoten Golf Links	Yes
Telenor	Lofoten Golf Links	Yes
Google	Lofoten Startside	Yes

[Add New](#) [First](#) [Prev](#) 1 of 1 [Next](#) [Last](#)

Add/Edit Advertisement

Company Id:
Country Id:
Position:
Title:
Image:
URL: (Must include http://)
Priority:
Online: ☐

Figur 17, Brukergrensesnitt – Add/Edit Banner

6.8.4 Add/Edit Internal Link

Denne funksjonaliteten gjør det mulig for administratoren å legge inn nye linker som vises på informasjonssystemet. Dette er enda en modul som bidrar til at informasjonssystemet er 100 % webbasert og at alle endringer kan gjøres fra hvilken som helst arbeidsstasjon, hvor som helst i verden.

List of Internal Link

Total Records: 8

Link Title
Blog
Forum
User Controlpanel
Admin Controlpanel
Search through archive
Forgotten your password
Search Provider
Register as new user

[Add New](#) [First](#) [Prev](#) 1 of 1 [Next](#) [Last](#)

Add/Edit Internal Link

Link Title:
Link Url:

Figur 18, Brukergrensesnitt – Add/Edit Internal Link

6.9 Sub systemer

Som en del av kravspesifikasjonen kom de tydelig fram at informasjonssystemet må inkludere mest mulig ferdig utviklede tjenester. Valget falt på forum- og blogg applikasjonen. Dette er applikasjoner som brukes av andre portaler og brukerne sikkert vil kjenne igjen og venne seg lett til dem. Forum er utviklet av phpBB⁵³, og bloggen er utviklet av bBlog⁵⁴. I dette dokumentet vil jeg ikke beskrive detaljert disse sub systemene eller deres bruksmønstre. Jeg har gjenbrukt en del av databasetabeller disse applikasjonene baserer seg på. Dette vil bli beskrevet lengre nede i besvarelsen. Viktigheten av disse to sub systemene er enorm når man ser det hele i en sammenheng. Disse applikasjonene gjør det mulig å ha interaktivitet mellom brukerne på mange varierte måter, samt at de er spekket med funksjonalitet som lett kan gjenbrukes av andre moduler.

6.10 Detaljerte usecase beskrivelser

Hver usecase som er nevnt tidligere i teksten består av tre moduler: Legge til / Endre / Slette. Jeg betrakter disse som en helhet når jeg beskriver overordnet usecase, derfor beskrives disse underfunksjoner sammen. Se, *relevant informasjon*.

6.10.1 Add/Edit Article

Usecase	Add/Edit Article (Legg til/Endre artikkel)
Trigger:	Det skal legges inn en ny artikkel
Aktør:	Traveler og Provider
Prebetingelser:	Aktøren vet hvilken informasjon skal legges inn, samt at et bilde legges ved
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. User Controlpanel 2. Add/Edit Article 3. Skriv inn informasjon, velg et bilde som skal legges ved 4. Legg informasjonen i database – Add 5. Velg artikkel fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom artikkelen skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 3.2 Bildet er ikke lagt ved – systemet ber om rettelse 3.2 Bildet er for stort – systemet ber om rettelse 7.1 Manglende informasjon – systemet ber om rettelse
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 6, Usecase beskrivelse – Add/Edit Article

6.10.2 Add/Edit Link

Usecase	Add/Edit Link (Legg til/Endre link)
Trigger:	Det skal legges inn en ny link relatert til artikkel
Aktør:	Traveler og Provider
Prebetingelser:	Aktøren vet hvilken informasjon skal legges inn, samt at en artikkel allerede er lagt inn i systemet
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. User Controlpanel 2. Add/Edit Link 3. Skriv inn informasjon, velg artikkel det skal legges til 4. Legg informasjonen i database – Add 5. Velg link fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom link skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 3.2 Artikkel er ikke valgt – systemet ber om rettelse 7.1 Manglende informasjon – systemet ber om rettelse
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 7, Usecase beskrivelse – Add/Edit Link

6.10.3 Add/Edit Attachment

Usecase	Add/Edit Attachment (Legg til/Endre vedlegg/bilde)
Trigger:	Det skal legges inn en nytt vedlegg/bilde relatert til artikkel
Aktør:	Traveler og Provider
Prebetingelser:	Aktøren vet hvilke vedlegg/bilder skal legges inn, samt at en artikkel allerede er lagt inn i systemet
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. User Controlpanel 2. Add/Edit Attachment 3. Velg fil for opplasting, velg artikkel det skal legges til 4. Last opp fil på server, legg informasjon i databasen – Add 5. Velg fil fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom fil skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 3.2 Artikkel er ikke valgt – systemet ber om rettelse 7.1 Manglende informasjon – systemet ber om rettelse
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 8, Usecase beskrivelse – Add/Edit Link

6.10.4 Add/Edit Comment

Usecase	Add/Edit Comment (Legg til/Endre kommentar)
Trigger:	Det skal legges inn en nytt kommenter på bildet eller artikkel
Aktør:	Traveler og Provider
Prebetingelser:	Aktøren vet hvilke informasjon skal legges inn. Funksjonen er startet fra et allerede eksisterende artikkel eller bilde som vises på informasjonssystemet
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. User Controlpanel 2. Add/Edit Comment 3. Skriv inn informasjon som skal legges inn 4. Legg informasjon i databasen – Add 5. Velg kommentar fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom kommentar skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 3.2 Funksjonalitet er startet opp med feil parameter – systemet ber om rettelser 7.1 Manglende informasjon – systemet ber om rettelse 7.2 Brukeren er ikke opphavsmann til kommentaren – Error 8.1 Brukeren er ikke opphavsmann til kommentaren – Error
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 9, Usecase beskrivelse – Add/Edit Comment

6.10.5 Search Provider

Usecase	Search Provider (Søk etter tilbydere)
Trigger:	Det skal søkes etter tilbydere
Aktør:	Traveler, Provider og Admin
Prebetingelser:	Aktøren vet hvilke kriterier skal det søkes etter
Postbetingelser:	1. Databasen inneholder register av tilbydere 2. Systemet viser informasjonene fra registeret
Normal hendelsesflyt:	1. Search Provider 2. Alle tilgjengelige tilbydere listes opp 3. Skriv inn kriterier – Search 4. Informasjon basert på kriterier listes opp
Variasjoner:	2.1 Ingen tilbydere registrert 4.1 Informasjon basert på kriterier finnes ikke
Relevant informasjon:	Søk med blanke søkefelt lister opp alle tilbydere. Det kan søkes med flere kriterier samtidig.

Tabell 10, Usecase beskrivelse – Search Provider

6.10.6 Search Archive

Usecase	Search Archive (Søk i arhiv)
Trigger:	Det skal søkes etter registrert informasjon
Aktør:	Traveler, Provider og Admin
Prebetingelser:	Aktøren vet hvilke kriteria skal det søkes etter
Postbetingelser:	1. Databasen er ikke tom 2. Systemet viser informasjonene fra databasen
Normal hendelsesflyt:	1. Search Archive 2. All tilgjengelig informasjon om <i>artikler</i> , <i>bokmerker</i> og <i>vedlegg</i> vises 3. Skriv inn kriteria – Search 4. Informasjon basert på kriteria listes opp
Variasjoner:	2.1 Ingen informasjon i databasen 4.1 Informasjon basert på kriteria finnes ikke
Relevant informasjon:	Søk med blanke søkefelt lister opp alle artikler. Det kan søkes med flere kriteria samtidig. Søkefeltene <i>title</i> og <i>tags</i> er felles for alle tre kategorier.

Tabell 11, Usecase beskrivelse – Search Archive

6.10.7 Edit Article

Usecase	Edit Article (Endre artikkel)
Trigger:	Det er nødvendig å endre innhold i en artikkel eller slette den
Aktør:	Admin
Prebetingelser:	Aktøren vet hvilke informasjon skal endres, samt hvilken artikkel som inneholder den informasjonen.
Postbetingelser:	1. Ønsket informasjon blir endret 2. Systemet oppdateres
Normal hendelsesflyt:	1. Admin Controlpanel 2. Edit Article 3. Velg artikkel fra listen 4. Endre informasjonen 5. Lagre endringer med – Submit 6. Dersom artikkel skal slettes permanent - Delete
Variasjoner:	4.1 Manglende informasjon – systemet ber om rettelser
Relevant informasjon:	

Tabell 12, Usecase beskrivelse – Edit Article

6.10.8 Add/Edit Company (provider)

Usecase	Add/Edit Company (provider) (Legg til/Endre tilbyder)
Trigger:	Det skal legges inn en tilbyder av reiser eller opplevelser
Aktør:	Admin
Prebetingelser:	Aktøren vet hvilke informasjon skal legges inn.
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. Admin Controlpanel 2. Add/Edit Comment 3. Skriv inn informasjon som skal legges inn 4. Legg informasjon i databasen – Add 5. Velg tilbyder fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom kommentar skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 7.1 Manglende informasjon – systemet ber om rettelse
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 13, Usecase beskrivelse – Add/Edit Company (provider)

6.10.9 Add/Edit Banner

Usecase	Add/Edit Banner (Legg til/Endre banner)
Trigger:	Det skal legges inn en ny banner som skal senere brukes for markedsføring på av tilbyderer på informasjonssystemet
Aktør:	Admin
Prebetingelser:	Aktøren vet hvilke banner skal legges inn, samt at tilbyderer allerede er lagt inn i systemet
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. Admin Controlpanel 2. Add/Edit Banner 3. Velg banner for opplasting, velg tilbyder det skal legges til 4. Last opp fil på server, legg informasjon i databasen – Add 5. Velg fil fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom fil skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 3.2 Tilbyder er ikke valgt – systemet ber om rettelse 7.1 Manglende informasjon – systemet ber om rettelse
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 14, Usecase beskrivelse – Add/Edit Banner

6.10.10 Add/Edit Internal Link

Usecase	Add/Edit Internal Link (Legg til/Endre intern link)
Trigger:	Det skal legges inn en ny intern link i menyen
Aktør:	Admin
Prebetingelser:	Aktøren vet hvilken informasjon skal legges inn, samt at en artikkel allerede er lagt inn i systemet
Postbetingelser:	1. Ønsket informasjon blir lagt til 2. Systemet oppdateres
Normal hendelsesflyt:	1. Admin Controlpanel 2. Add/Edit Internal Link 3. Skriv inn informasjon 4. Legg informasjonen i database – Add 5. Velg link fra listen 6. Endre informasjonen 7. Lagre endringer med – Submit 8. Dersom link skal slettes permanent - Delete
Variasjoner:	3.1 Manglende informasjon – systemet ber om rettelser 7.1 Manglende informasjon – systemet ber om rettelse
Relevant informasjon:	Systemet er robust og all Manglende informasjon blir varslet. Dette betyr at innkonsistent informasjon legges aldri i databasen. Punktene 1-4 beskriver den delen av funksjonaliteten som tar seg av innleggelsen av informasjonen. Restene av punktene beskriver brukstilfelle når brukeren vil endre eller slette informasjonen permanent.

Tabell 15, Usecase beskrivelse – Add/Edit Internal Link

6.11 Plattform og utviklingsverktøy

Under vurdering av plattformen applikasjonen skal bygges på, falt valget enkelt for PHP og MySQL kombinasjonen. Sub systemene som er nevnt bygger også på disse to teknologiene. Selv har jeg programmeringserfaring i PHP og MySQL, og jeg ville gjerne bruke min eksisterende kunnskap. Det er flere fordeler med å bruke disse to teknologier.

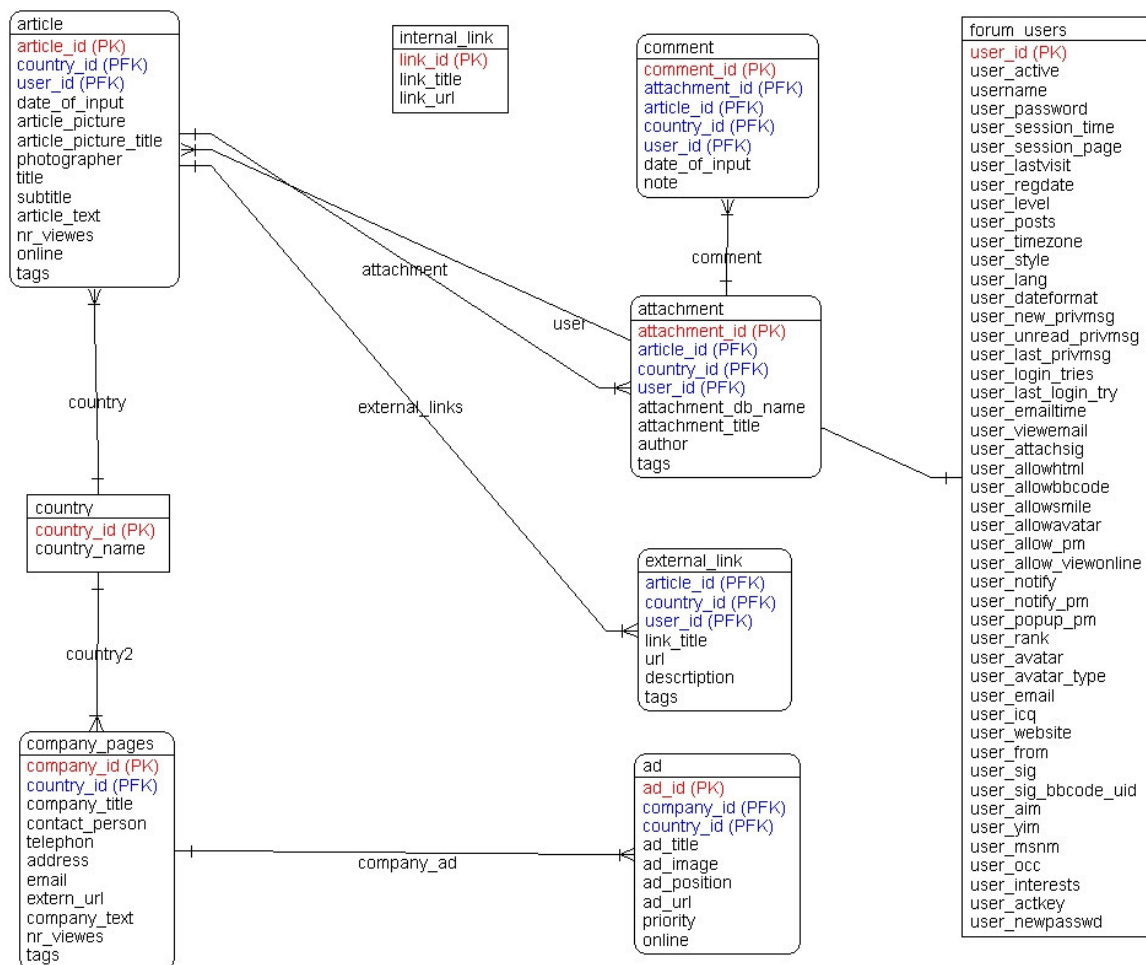
PHP var et verktøy for utvikling av hjemmesider med relativt enkel funksjonalitet. Det har imidlertid utviklet seg til et av de raskeste og mer avanserte verktøyene for store nettsteder. I motsetning til andre webutviklingsspråk som ASP og Java, er PHP laget spesielt for web. Språket tillater objektorientert programmering og har en god integrasjon mot forskjellige databaser. PHP eksekverer raskt og tåler høy trafikk. Sikkerheten er oversiktlig og bra. PHP kan kjøres både på Linux, Windows og andre plattformer. Websteder i PHP kan utveksle data med websteder utviklet med annen teknologi via XML. PHP brukes i dag til store, kritiske systemer som for eksempel børshandel over internett.

MySQL er en meget rask database for lagring og henting av data. Innholdet i databaser vil også ofte flettes inn i websider, samt styre utseendet og funksjoner på siden. PHP bruker ofte MySQL databaser til dette formålet. MySQL passer for små og mellomstore databaser. Hvis MySQL kjører på Linux, tåler den store databaser med høy trafikk.

PHP og MySQL er gratis Open Source verktøy. Som webserver benyttes vanligvis Apache. Denne kombinasjonen gir en meget rask webtjeneste. Kildekoden til alle Open Source verktøy er fritt tilgjengelig. Det samme er en rekke applikasjoner og bibliotek for PHP og MySQL. Eventuell support kan kjøpes fra mange konkurrerende leverandører. Et nettsted utviklet i PHP og MySQL kan enkelt utplasseres og driftes hos en ISP. Siden softwaren er gratis, er dette normalt den rimeligste webhostingen som finnes.

6.11.1 ER diagram – Authentic Travel

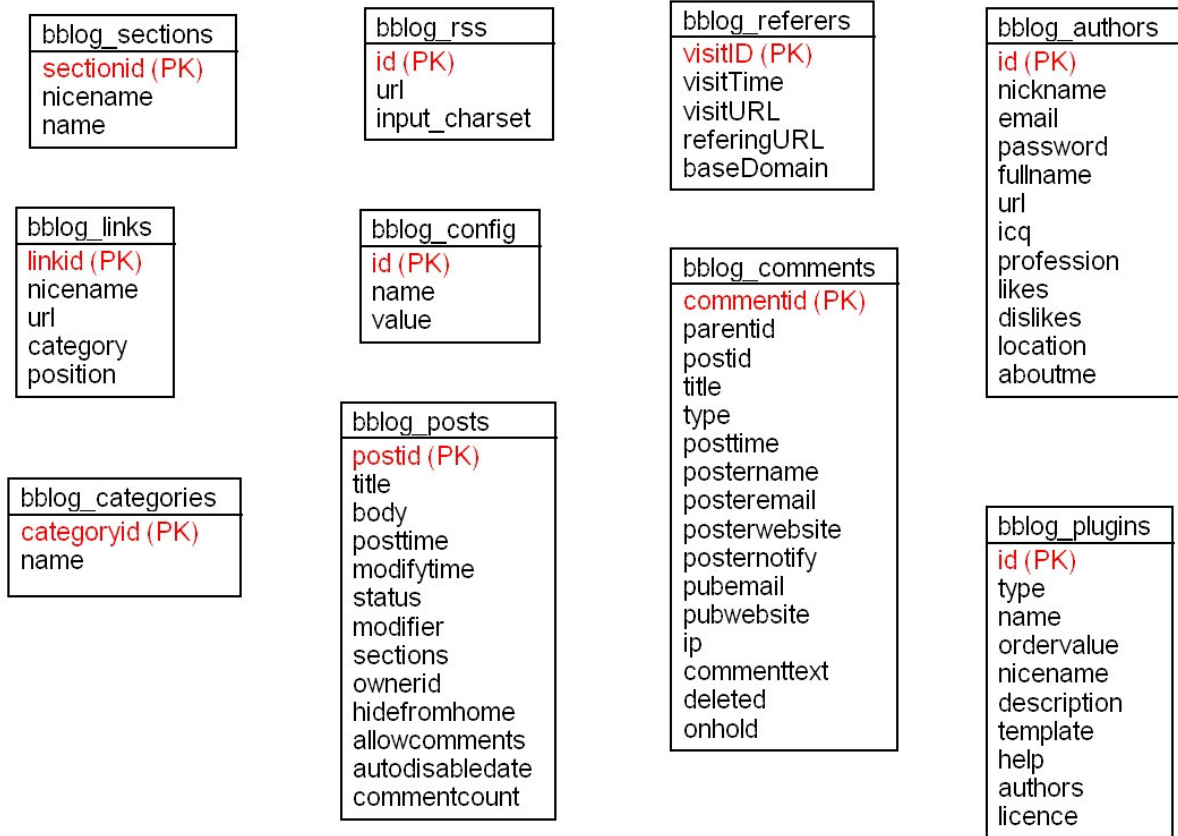
Informasjonen som skapes på informasjonssystemet vil bearbejdes av PHP, lagres i - og hentes fra MySQL databasen. Databasen må klargjøres for mottak av data. De kommende tre ER diagrammene⁵⁵ beskriver strukturen på databasen som lagrer data brukt av systemet. Dette første ER diagrammet er det mest vesentlige med tanken på den løsningen jeg har kommet med. Tabellene dataene lagres i og relasjonene mellom dem er skreddersydd for dette formålet. *forum_user* tabellen er lånt fra gruppen tabeller som definerer sub systemet *Forum*. Forum systemet har en rekke mekanismer for håndtering av dataene i denne tabellen, så jeg valgte å bruke disse.



Figur 19, ER diagram – Authentic Travel

6.11.2 ER diagram – Blogg

Dette ER diagrammet viser tabellene som brukes i sub systemet *Blogg*. Blogg, eller weblogg, blir sidene organisasjonen Authentic Travel legger ut informasjon som andre kan kommentere. Det er mening at det skal brukes som logg eller dagbok skrevet i en uformell stil.

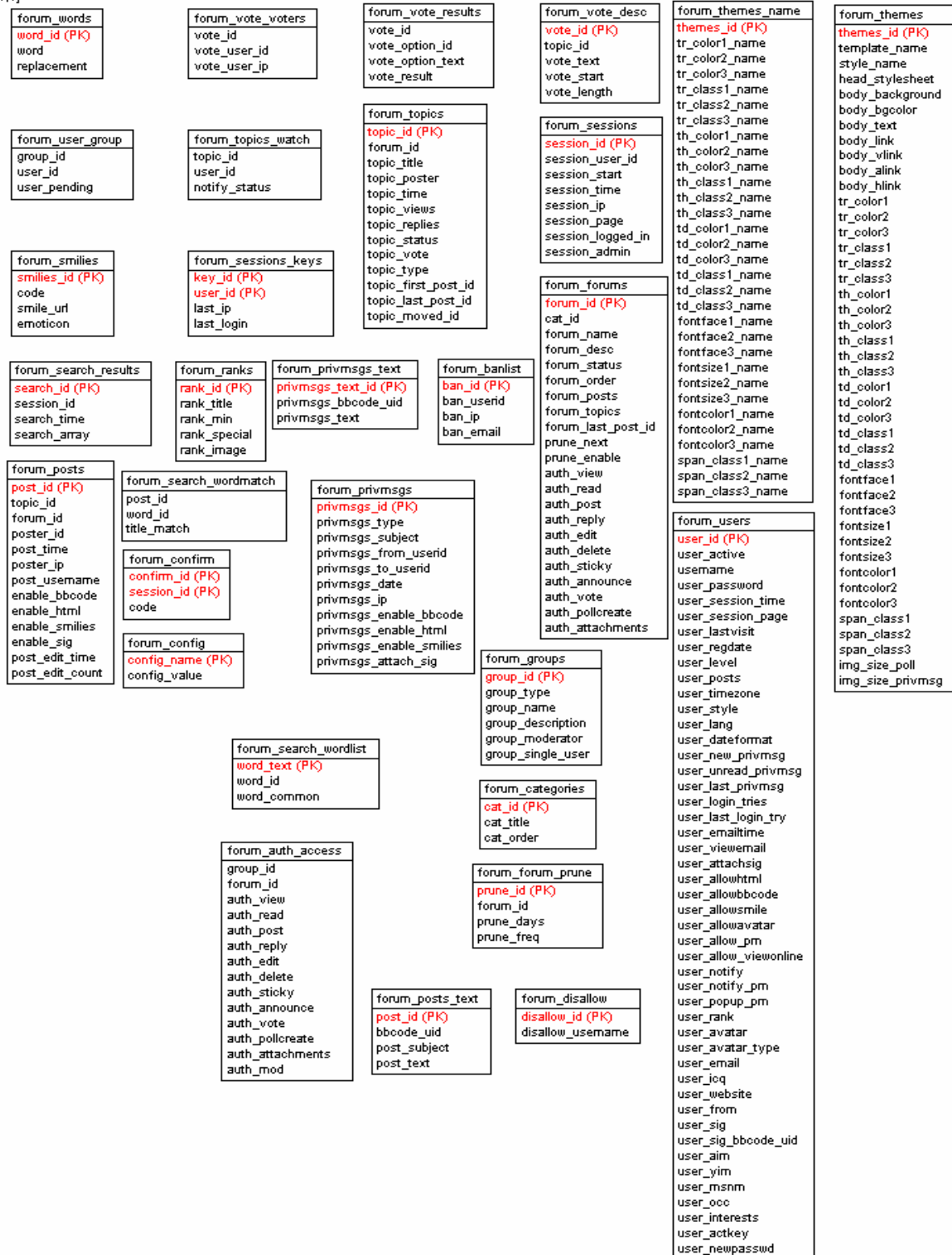


Figur 20, ER diagram – blogg

6.11.3 ER Diagram – Forum

Informasjonssystemet vil også basere seg på diskusjonsforum. Dette skal bli arena der brukerne diskuterer fritt. Blogg vs. Diskusjonsforum: forskjellen er mellom formell og uformell stil. Jeg velger å ta begge i bruk, blant annet på grunn av måten innleggene organiseres. Blogg vil være forbeholdt organisasjonen, mens forum er for brukerne.

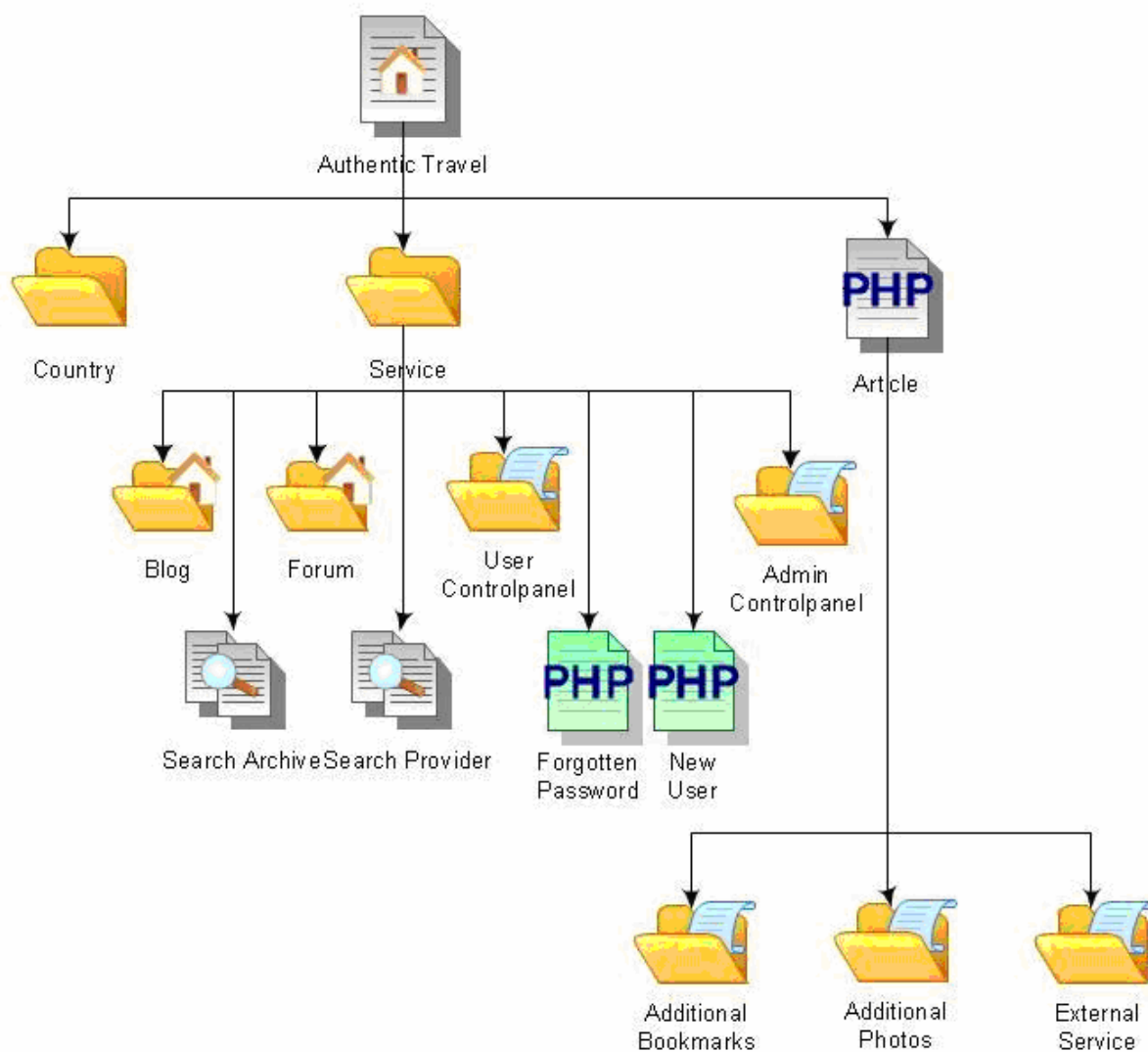
[1,1]



Figur 21, ER diagram – forum

6.12 Sidestruktur (3-nivå)

Denne modellen skisserer oss strukturen på websiden. Modellen går ikke dypt i strukturen, så vi ser bare en overordnet struktur som er tre nivåer dyp.



Figur 22, Sidestruktur – 3 nivå

6.13 SQL

Structured Query Language (SQL) er et språk som benyttes til å formulere og kjøre operasjoner mot databaser. De fleste av dagens databasesystemer tilbyr SQL som kontrollgrensesnitt. Jeg har brukt SQL til å få fram informasjon fra databasen, så brukt den til å bygge opp informasjonssystemet. Disse spørresetninger framstilles her uten all den relevante koden som ligger rundt dem og som gjør at informasjon vises på websidene. Som vedlegg til dette dokumentet ligger det en CD med alle filer som inngår i dette informasjonssystemet. Nærmere titt på innhold i disse filene avslører hvordan disse spørresetningene er implementert. Inputparametrer som spørresetninger bruker kan være URL- eller session variabler. De beskrives som `{parameter}`.

6.13.1 Liste artikler

```
SELECT article.*, country_name
FROM article, country
WHERE article.online = '{online}'
AND article.country_id = country.country_id
ORDER BY article_id desc
```

Denne spørringen brukes på den første siden og den genererer liste over artikkler som er lagt inn i databasen. Listen er sortert etter synkende rekkefølge. Da, vises den siste innlagte artikkel øverst

6.13.2 Liste land

```
SELECT country_name, country.country_id AS
country_country_id
FROM article, country
WHERE article.country_id = country.country_id
GROUP BY country_name
ORDER BY country_name
```

Denne spørringen brukes til å liste navn på land som knyttes til artikkler som er lagt inn i databasen. Navnene er gruppert, slik at flere like navn ikke vises. De er også sortert alfabetisk. Denne listen vises på alle sidene.

6.13.3 Liste tjenester

```
SELECT *
FROM internal_link
```

Denne spørringen brukes til å liste URL navn og adresser til interne tjenester knyttet til systemet.

6.13.4 Liste de 10 siste fra blogg

```
SELECT *
FROM bblog_posts
ORDER BY postid desc
```

Denne spørringen brukes til å liste de ti siste innleggene i sub systemet blogg. Listen er sortert slik at de nyeste innleggene vises øverst.

6.13.5 Liste de 10 siste fra forum

```
SELECT forum_posts.topic_id AS  
forum_posts_topic_id, topic_title  
FROM forum_posts, forum_topics  
WHERE forum_posts.topic_id =  
forum_topics.topic_id  
GROUP BY forum_posts.topic_id  
ORDER BY forum_posts.topic_id desc
```

Denne spørringen brukes til å liste de ti siste innleggene i sub systemet forum. Listen er sortert slik at de siste innleggene vises øverst.

6.13.6 Liste artikler knyttet til et bestemt land

```
SELECT article.*, country_name  
FROM article, country  
WHERE article.country_id = {country_id}  
AND article.country_id = country.country_id  
AND article.online = '{online}'  
ORDER BY article_id desc
```

Denne spørringen brukes til å liste artikler som knyttes til et bestemt land. Spørringen er videreutviklet fra den beskrevet i punktet 6.13.1. Listen av artikler som vises tilhører et bestemt land. Behøver inputparameter.

6.13.7 Liste eksterne linker (bokmerker)

```
SELECT *  
FROM external_link  
WHERE article_id = {article_id}
```

Denne spørringen brukes til å liste vedlagte bokmerker til en bestemt artikkel. Behøver inputparameter.

6.13.8 Liste vedlegg (bilder)

```
SELECT *  
FROM attachment  
WHERE article_id = {article_id}
```

Denne spørringen brukes til å liste vedlagte filer (bilder) til en bestemt artikkel. Behøver inputparameter.

6.13.9 Eksterne tjenester

```
SELECT tags  
FROM article  
WHERE article_id_random = {article_id_random}  
AND article_id = {article_id}
```

Denne spørringen brukes til å generere informasjon (tags) fra en bestemt artikkel. Disse taggene flettes da med en URL adresse til en ekstern tjeneste.

6.13.10 **Vise artikkel**

```
SELECT country_name, date_of_input,  
article_picture, article_picture_title, photographer,  
title, subtitle, article_text, tags, username,  
article.user_id AS article_user_id  
FROM article, coutry, forum_users  
WHERE article.article_id_random =  
{article_id_random}  
AND article.user_id = forum_users.user_id  
AND article.country_id = country.country_id  
AND article.article_id = {article_id}  
AND article.online = '{online}'
```

Denne spørringen brukes til å vise mesteparten av informasjonen som knyttes til en artikkel. Systemet krever flere inputparametrer for at artikkelen skal vises. Det må være en riktig sammenheng mellom disse, ellers vises ikke artikkelen.

6.13.11 **Liste kommentarer (artikkel)**

```
SELECT comment.*, username  
FROM comment, forum_users  
WHERE comment.article_id = {article_id}  
AND comment.user_id = forum_users.user_id  
AND comment.country_id = {country_id}
```

Denne spørringen brukes til å liste kommentarer knyttet til en bestemt artikkel. Spørringene krever riktige inputparametrer for å liste informasjon.

6.13.12 **Liste kommentarer (bilde)**

```
SELECT comment.*, username  
FROM comment, forum_users  
WHERE comment.country_id = {country_id}  
AND comment.user_id = forum_users.user_id  
AND comment.attachment_id = {attachment_id}
```

Denne spørringen brukes til å liste kommentarer knyttet til et bestemt bilde som er lagt som vedlegg til en artikkel. Krever riktige inputparametrer.

6.13.13 **Vise banner (annonse)**

```
SELECT *  
FROM ad  
WHERE ad_position = '{ad_position}'  
AND online = '{online}'  
ORDER BY priority desc
```

Denne spørringen brukes til å liste annonsebannere. Spørringen brukes på tre forskjellige posisjoner på en side. Annonsebannere sorteres etter prioritet

6.14 Forretningsmodell

Mens utviklingen av funksjonaliteten har foregått i en forrykende fart, så har det stått stille når det gjelder forretningsmodellen knyttet til dette prosjektet. Slik utvikling hadde også tidligere beskrevne tilfeller (se kapittel Casestudier). Grunnen til dette er den tjenesteorienterte tankegangen bak slike applikasjoner. Med dette mener jeg at det dannes først store behov for å utvikle en applikasjon som skal tjene folket og deres behov. Etter hvert innser man at det er mulig å tjene penger på et større antall brukere, og annonseinntekter er det første og enkleste man tenker på. En annen utvei er å tenke oppkjøp. Håpe på at en større aktør på markedet skal innse verdien i tjenesten som er laget og kjøpe opp hele konseptet.

Jeg foreslår en forretningsmodell som baserer seg på annonseinntekter. Jeg har tilrettelagt funksjonalitet som gjør det enkelt å selge annonseplasser på sidene. Tilbydere av reiseopplevelser eller hvem som helst andre kan annonsere her, der prisen for disse tjenestene kan basere seg på flere inntjeningsmodeller:

- Antall genererte klikk
- Fast pris for annonse stående i en bestemt periode
- Profitt av kjøpet som sendt bruker har gjort hos annonsøren
- Ta i bruk Google sin AdSense annonseprogram

En annen inntjeningskilde er å kreve betaling for utvidet funksjonalitet som både aktøren Traveler og Provider kan få. Denne utvidede funksjonaliteten er ikke designet i dette prosjektet, men mulighetene for hva det kan bli er mange.

6.15 Sosiale effekter

Det er rimelig å tro at dette informasjonssystemet vil skape relasjoner mellom mennesker. Det blir mulig å se hvor andre har reist, hvorfor de er reist og hva de har opplevd på reisen. Autentiske reisedestinasjoner og små samfunn vil synliggjøres. I tillegg til at mennesker vil bli kjent med hverandre gjennom systemet, vil systemet også bidra til utvikling av lokalsamfunn som knyttes til autentiske reise destinasjoner.

7 Eksperimenter

Etter at informasjonssystemet ble utviklet, så har det blitt installert på en midlertidig webserver. Adressen er: <http://folk.uio.no/alence/> Jeg har selv utført en rekke tester som forsikret meg at den funksjonaliteten som er implementert fungerer tilfredsstillende. En rekke feil og mangler har blitt oppdaget gjennom dette, og de er blitt utbedret.

Jeg ville ha tilbakemeldinger som er mer varierte enn mine subjektive meninger. Derfor opprettet jeg en spørreundersøkelse og inviterte testbrukere. Testbrukerne er informatikkstudenter. Jeg har sendt flere massemail til studenter og bedt dem teste ut systemet. Jeg har for det meste fått positiv tilbakemelding om systemet, men det forekommer også en del konstruktive kritikker.

Som en del av de konstruktive kritikkene kom det tips om å bruke *stressverktøyet* JMeter. Dette verktøyet simulerer flere virtuelle brukere som samtidig benytter seg av systemet. Resultater av disse målinger viser robustheten som systemer har overfor høy brukertrafikk. Grunnet problemer med å sette opp JMeter applikasjonene riktig og få den til å utføre tester, har det ennå ikke blitt gjort noen målinger. Disse testene kan inngå i framtidig arbeid med systemet.

Det som med sikkerhet blir en verdifull eksperiment er planene for et realistisk prøveprosjekt på Hordaland i 2007. Systemet som er utviklet skal offentliggjøres og publikum skal kunne ta det i bruk. Tilbakemeldingene som da kommer vil danne utgangspunkt for videre utvikling av systemet. I god Web 2.0 ånd så vil systemet utvikle seg i den retning brukerne ønsker. I dag er systemet i betafasen og dermed tolkes å være fortsatt under utvikling.

I forbindelsen med eksperimenter samt videre arbeid anbefaler jeg på det sterkeste gode backup rutiner. Backup er prosessen man har for å kunne komme tilbake til en tilstand som er så nær situasjonen var før et totalt krasj. En backup løsning tar kopier av gitte data på et eksternt medium, for eksempel CD. Dette skjer som regel en gang daglig og kan settes opp etter behov. Dersom uhellet skulle inntreffe har man da muligheten til å hente tilbake de data som man har kopiert ut da backup prosessen gikk. Dette vil spare en for enormt mye arbeid hvis det skulle skje noe uforutsigbart.

8 Videre arbeid

Det er en del videre arbeid med informasjonssystemet som er utviklet. Selv om systemet har funksjonelle søkemekanismer, gjenstår det mye arbeid med å implementere mekanismer som skal ordne opp i informasjonskaoset som vil oppstå med tiden

- Slå sammen *tilbud* og *etterspørsel* på en effektiv måte.
- Organisere den store mengden informasjon
- Systemet er designet for slikt bruk at det produseres mye informasjon
- Hvordan holde oversikt over informasjonene
- Hvordan ta i bruk mekanismer som gjør det mulig å finne relevant informasjon

Det vil bli uakseptabelt å komme i en situasjon der systemet besitter stor mengde nyttig informasjon, men at denne vises ikke når brukeren trenger det mest. Som en løsning på dette vil *Topic Maps*⁵⁶ bli implementert. Denne jobben vil bli gjort i forbindelse en masteroppgave, eller at det folk med kompetanse påtar seg jobben.

En del funksjonalitet på selve sidene bør også lages. Da tenker jeg på implementering av:

- Rangeringsmodul
- Stjernemodul
- Diverse tellere som måler trafikken på sidene
- Avstemningsmoduler

Som tidligere nevnt, vil prøveprosjekt basert på informasjonssystemet som er laget finne sted i Hordaland. Det videre arbeidet med systemet vil mye handle om dette. Det er rimelig å tenke seg at en del jobb må gjøres både før og etter dette prøveprosjektet.

9 Konklusjon

Det skapes mange nye trender og sosiale samspill som et resultat av Web 2.0 teknologier, men fortsatt kommer det ikke mye økonomisk utbytte ut av det. Foretningsmodeller er fortsatt vanskelig å definere, og det er lett for Web 2.0 bedrifter å ignorere dette. Situasjonen minner oss om den første internett boblen som fant sted på slutten av 90-tallet, og det er ikke overraskende at analytikere kaller dette Boble 2.0 i stedet for Web 2.0. Tross dette, ser vi at noen teknologier og bedriften har allerede oppstått fra denne *boblen*, og har skapt levedyktige forretninger. Det er mange flere teknologier og bedrifter som prøver på det samme. Samtidig er en stor interesse fra investorer, mye forskning og initiativ fra folk som gjøres nå om dagen for at disse teknologier og bedrifter skal bli enda bedre og overleve i framtiden.

Selv om det finnes skepsis, er det ikke vanskelig å se at denne samlingen av teknologier og forretningsmodeller, kjent som Web 2.0, kommer til øke i popularitet. En grunn til dette er at de baseres seg på enkle prinsipper. I tillegg, viser det seg at disse enkle teknologier som brukes til å lage sosiale nettverk er gunstig plattform for å skape forretninger. Mange av disse nøkkelteknologier har vært tilgjengelige på markedet i forskjellige former i lang periode før Web 2.0 trenden. Det er veldig liten teknisk innovasjon som har skjedd, men bare måtene de brukes på og sammenhengene de finnes i er helt nye. Mulighetene for større lagring, raskere servere, større båndbredde og ikke minst en økende vilje blant utviklere til stole på brukere til å skape innhold og gi tilbakemeldinger har vært en stor pådriver i utviklingen som fortsatt er akselererende. Web 2.0 er et symbol på sammensmelting av muligheter for forretninger, tekniske nyvinninger og sosiale samspill. For tiden virker det som om den største vekten blir lag på det sosiale aspektet. Hvis mulighetene brukes riktig, så kan det dannes gode muligheter for levedyktige forretninger, og ikke bare enkle *elektroniske samfunn*.

Nye Web 2.0 bedrifter må skapes med omhu, siden forretningsmodeller forblir vanskelige å definere. Hindringene for oppstart er ganske lave, siden det ennå ikke eksisterer dominante aktører på markedet. Dette vil si at alle kan prøve seg og like mange kan feile. Det meste av dagens teknologier er lettilgjengelige og gratis å bruke. Dette fører til lave kostnader for Web 2.0 bedrifter. Hvis en Web 2.0 bedrift skal overleve, så må den differensiere seg fra andre, og med det ta større markedsandeler og antall brukere. Fordelen ved å være først eller størst vs. sist og minst, vil utgjøre forskjellen mellom suksess og ikke – suksess. Det negative med å ikke være først eller størst er at bedrifter kommer i en liten og mettet nisjemarked, der det gjerne er plass til bare få aktører – og de siste som har entret nisjemarkedet vil ikke lykkes. Uten en sterk differansefaktor i verken teknologi eller service, blir det vanskeligere å engasjere mange nok brukere for å opprettholde en Web 2.0 bedrift. Det vil bli bare noen få rene og suksessrike Web 2.0 bedrifter. Mindre Web 2.0 bedrifter implementerer sine teknologier i bedrifter med markedsandel som pr. dags dato ikke er definert som Web 2.0 bedrifter. Med disse oppdateringer blir tradisjonelle bedrifter omformet til suksessfulle Web 2.0 bedrifter. Attraktive oppkjøp av store internettaktører er blitt rent symbol på suksessen en Web 2.0 bedrift kan oppnå. Disse oppkjøp har relatert seg til bedrifter som har klart å mobilisere stort antall lojale brukere som benytter seg av deres Web 2.0 applikasjoner.

Vedlegg

Her kommer stoffet som ble for detaljert til å bli med i selve rapporten. Programkoden med mye mer er omfattende og alt ligger på en vedlagt CD. Jeg velger å legge ved DDL script som er relevant til ER diagram beskrevet på Figur 19, ER diagram – Authentic Travel. Scriptet brukes til å opprette tabeller i databasen som skal brukes med systemet.

```
--
-- Table structure for table `ad`
--

CREATE TABLE `ad` (
  `ad_id` int(11) NOT NULL auto_increment,
  `company_id` int(11) NOT NULL,
  `country_id` int(11) NOT NULL,
  `ad_title` varchar(96) NOT NULL,
  `ad_image` varchar(96) NOT NULL,
  `ad_position` varchar(20) NOT NULL,
  `ad_url` varchar(255) NOT NULL,
  `priority` int(11) NOT NULL default '0',
  `online` char(3) NOT NULL,
  PRIMARY KEY (`ad_id`,`company_id`,`country_id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;

--
-- Table structure for table `article`
--

CREATE TABLE `article` (
  `article_id_random` int(15) NOT NULL,
  `article_id` int(11) NOT NULL auto_increment,
  `country_id` int(11) NOT NULL,
  `user_id` mediumint(9) NOT NULL,
  `date_of_input` varchar(50) character set utf8 collate utf8_unicode_ci
NOT NULL,
  `article_picture` varchar(60) NOT NULL,
  `article_picture_title` varchar(60) NOT NULL,
  `photographer` varchar(60) NOT NULL,
  `title` varchar(60) NOT NULL,
  `subtitle` varchar(160) NOT NULL,
  `article_text` text NOT NULL,
  `online` char(3) NOT NULL,
  `tags` varchar(160) NOT NULL,
  `nr_views` int(11) NOT NULL,
  PRIMARY KEY (`article_id`,`country_id`,`user_id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;

--
-- Table structure for table `attachment`
--

CREATE TABLE `attachment` (
  `attachment_id` int(11) NOT NULL auto_increment,
  `article_id` int(11) NOT NULL,
  `country_id` int(11) NOT NULL,
```

```
`user_id` mediumint(9) NOT NULL,  
`attachment_db_name` varchar(200) NOT NULL,  
`attachment_title` varchar(60) NOT NULL,  
`author` varchar(60) default NULL,  
`tags` varchar(160) NOT NULL,  
PRIMARY KEY (`attachment_id`,`article_id`,`country_id`,`user_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;
```

```
--  
-- Table structure for table `comment`  
--
```

```
CREATE TABLE `comment` (  
  `comment_id` int(11) NOT NULL auto_increment,  
  `attachment_id` int(11) NOT NULL,  
  `article_id` int(11) NOT NULL,  
  `country_id` int(11) NOT NULL,  
  `user_id` mediumint(9) NOT NULL,  
  `date_of_input` varchar(30) NOT NULL,  
  `note` text NOT NULL,  
  PRIMARY KEY  
  (`comment_id`,`attachment_id`,`article_id`,`country_id`,`user_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;
```

```
--  
-- Table structure for table `company_pages`  
--
```

```
CREATE TABLE `company_pages` (  
  `company_id` int(11) NOT NULL auto_increment,  
  `country_id` int(11) NOT NULL,  
  `company_title` varchar(96) NOT NULL,  
  `contact_person` varchar(50) NOT NULL,  
  `telephon` varchar(20) NOT NULL,  
  `address` varchar(96) NOT NULL,  
  `email` varchar(96) default NULL,  
  `extern_url` varchar(200) default NULL,  
  `company_text` text,  
  `nr_views` int(11) NOT NULL,  
  `tags` varchar(160) NOT NULL,  
  PRIMARY KEY (`company_id`,`country_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

```
--  
-- Table structure for table `country`  
--
```

```
CREATE TABLE `country` (  
  `country_id` int(11) NOT NULL auto_increment,  
  `country_name` varchar(30) NOT NULL,  
  PRIMARY KEY (`country_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=236 ;
```

```
--  
-- Table structure for table `external_link`  
--
```

```
CREATE TABLE `external_link` (  
  `link_id` int(11) NOT NULL auto_increment,  
  `link_id_random` int(11) NOT NULL,  
  `article_id` int(11) NOT NULL,  
  `country_id` int(11) NOT NULL,  
  `user_id` mediumint(9) NOT NULL,  
  `link_title` varchar(160) default NULL,  
  `url` text,  
  `description` varchar(160) NOT NULL,  
  `tags` varchar(160) NOT NULL,  
  PRIMARY KEY  
  (`link_id`,`link_id_random`,`article_id`,`country_id`,`user_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=8 ;  
  
--  
-- Table structure for table `internal_link`  
--  
  
CREATE TABLE `internal_link` (  
  `link_id` int(11) NOT NULL auto_increment,  
  `link_title` varchar(160) NOT NULL,  
  `link_url` varchar(160) NOT NULL,  
  PRIMARY KEY (`link_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=14 ;  
  
--  
-- Table structure for table `pictures`  
--  
  
CREATE TABLE `pictures` (  
  `picture_id` int(11) NOT NULL auto_increment,  
  `article_id` int(11) NOT NULL,  
  `country_id` int(11) NOT NULL,  
  `user_id` mediumint(9) NOT NULL,  
  `picture_db_name` varchar(200) NOT NULL,  
  `picture_title` varchar(60) NOT NULL,  
  `photographer` varchar(60) NOT NULL,  
  PRIMARY KEY (`picture_id`,`article_id`,`country_id`,`user_id`),  
  KEY `IX_extra_pictures` (`article_id`,`country_id`,`user_id`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

Stikkordregister

A

AdSense;23
Ajax;8
AJAX;18
Analyse;34
Apache;12; 20
API;17; *Se* Mashups
Atom;8
Autentisk reising;36
Authentic Travel;3

B

Backup;69
bBlog;50
beta;12
Beta;10
blogg;7
bloggere. *Se* blogg
bokmerker;7
brukergrensesnitt;18
brukernettverkeffekter;14
Bubble 2.0;27; *Se* dotcom-boble

C

click-fraud;23

D

del.icio.us;15; 31
Design;34
dotcom-boblen;11
Drift og vedlikehold;34

E

ER diagram;61
Evolusjonsmodellen;35
Exitstrategier;26

F

Feeds. *Se* RSS, Atom
Flash;18
Flex. *Se* Flash
flickr;32
Flock;11
forretningsmodell;22
Fossefall/vannfallsmodellen;35

G

GMail;12; 18
Google Maps;8; 17; 22; 30

H

Hardware;20

I

Implementasjon;34
Informasjonssystemer;34
Infrastruktur;20
interaktivitet;7

J

JavaScript;8; 17; 18

K

Kod og fiks modellen;35
kulturell miljøvern;3

L

Laszlo. *Se* Flash
les-skriv nett;7
Linux;12; 15; 20
long tail market;25

M

Markedsføring;20
Mashups. *Se* API
Meme Map;2
Modelleringspråk;35
MySQL;41; 60

N

netthandel;1

O

O'Reilly Media;6
Open Source;60
OpenOffice.org;8
OPML;8

P

perpetual beta. *Se* beta
PHP;41; 60
phpBB;50
podcasts;7
Problembeskrivelse;34

R

RSS;8; 15

RSS feeds. *Se* RSS, feeds

S

SOAP;12; 17

Software;20

sosial interaksjon;7

sosiale nettverk;7

sosialnettverksider;6

Spiralmodellen;35

startups;11

stressverktøyet;69

T

tagging. *Se* tags

tags;7; 45

Topic Maps;70

transaksjonsbaserte systemer;34

U

use case;42

Uviklingsmodeller;35

V

videoblogging;7

W

Web 1.0;1; 7; 18

Web 1.5;1

Web 2.0;1; 2; 6

Wiki;8

Wikipedia;28; *Se* Wiki

World Wide Web;1

Writely;8

X

XML;8; 12; 17; 18

Referanser

- ¹ Websider som besøkende ikke går til og leser, men som besøkende utvikler og utvider selv. Websidene som realiserer samarbeid, samt personlig uttrykk. Eksempel dette er <http://myspace.com/>
- ² Wiki er en type webside som gjør det mulig for besøkende å legge til eller slette informasjon. Dette gjøres enkelt, på lik linje å endre allerede eksisterende informasjon. Det er lov å gjøre dette anonymt og uten registrering, men det finnes unntak for visse Wiki.
- ³ Fasilitet for å avholde diskusjon online.
- ⁴ Blogg er en sammenstilling av ordene Web og Logg. En blogg er online logg eller dagbok.
- ⁵ Tim O'Reilly er gründeren av O'Reilly Media. Han er tilhenger av bevegelsen for fri programvare og Open Source.
- ⁶ John Linwood Battelle er journalist. Han er kjent for sitt engasjement for Business 2.0 og Web 2.0.
- ⁷ Wikipedia, <http://wikipedia.org/>
- ⁸ Welcome to Flickr - Photo Sharing, <http://www.flickr.com/>
- ⁹ YouTube - Broadcast Yourself, <http://youtube.com/>
- ¹⁰ NRK.no – Podkast, <http://nrk.no/podkast/>
- ¹¹ Instant messaging, f.eks MSN, ICQ, og lignende.
- ¹² <http://spurl.net/> || <http://del.icio.us/>
- ¹³ <http://digg.com/> || <http://www.newsvine.com/>
- ¹⁴ Feedreader, http://www.download.com/3120-20_4-0.html?tg=dl-20&qt=feed%20reader&tag=srch
- ¹⁵ Apple - iPod + iTunes, <http://www.apple.com/itunes/>
- ¹⁶ Google Maps, <http://www.google.com/maphp?hl=en&tab=wl&q=>
- ¹⁷ Welcome to Writely!, <http://www.writely.com/>
- ¹⁸ Go2Web20.net - The complete Web 2.0 directory, <http://www.go2web20.net/>
- ¹⁹ Extensible Markup Language (XML), <http://www.w3.org/XML/>
- ²⁰ W3C XML Protocol Working Group, <http://www.w3.org/2000/xml/Group/>
- ²¹ The Linux Home Page at Linux Online, <http://www.linux.org/>
- ²² Welcome! - The Apache HTTP Server Project, <http://httpd.apache.org>
- ²³ Release Early, Release Often, <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/ar01s04.html>
- ²⁴ Wired News: Mo' Beta Testing Blues, <http://www.wired.com/news/infostructure/0,1377,63631,00.html>
- ²⁵ Welcome to Gmail, <http://www.gmail.com>
- ²⁶ JD on EP: Perpetual beta, http://weblogs.macromedia.com/jd/archives/2004/06/perpetual_beta.cfm
- ²⁷ Network effect on Wikipedia, http://en.wikipedia.org/wiki/Network_effect
- ²⁸ Slashdot: News for nerds, stuff that matters, <http://slashdot.org/>
- ²⁹ PenguinWebHosting.com - Affordable Linux Website Hosting, <http://www.penguinwebhosting.com/>
- ³⁰ RSS on Wikipedia, <http://en.wikipedia.org/wiki/Rss>
- ³¹ Google Maps API, <http://www.google.com/apis/maps/>
- ³² Adobe Flash Player, <http://www.macromedia.com/software/flash/about/>
- ³³ Adobe Flex is an IDE for building web applications and rich internet applications, <http://www.adobe.com/products/flex/>
- ³⁴ Laszlo Systems, <http://www.laszlosystems.com/>
- ³⁵ Sherwin Williams, http://sherlink.sherwin.com/swapp/color_visualizer/index.jsp
- ³⁶ JotSpot: enterprise and personal wiki hosting and free wiki software trial. Get a hosted wiki or download wiki software., <http://www.jot.com/>
- ³⁷ Open Source on Wikipedia, http://en.wikipedia.org/wiki/Open_source
- ³⁸ chicagocrime.org: Chicago crime database, <http://www.chicagocrime.org/>
- ³⁹ Google AdSense, <https://www.google.com/adsense/>
- ⁴⁰ Click Fraud on Wikipedia, http://en.wikipedia.org/wiki/Click_fraud
- ⁴¹ Croctech, http://www.jroller.com/page/croctech?entry=design_and_the_long_tail
- ⁴² PVMC: The Valuation Process Explained, http://www.pvmc.net/Valuation/Val_Process/val_process.html
- ⁴³ Bill Gates on Wikipedia, http://en.wikipedia.org/wiki/Bill_Gates
- ⁴⁴ Wiki on Wikipedia, <http://en.wikipedia.org/wiki/Wiki>
- ⁴⁵ Home - Wikimedia Foundation, <http://wikimediafoundation.org/>
- ⁴⁶ GNU Free Documentation License - GNU Project - Free Software Foundation (FSF), <http://www.gnu.org/copyleft/fdl.html>

- ⁴⁷ Housing Maps <http://www.housingmaps.com/>
- ⁴⁸ Google Content-Targeted Advertising FAQ, https://adwords.google.com/select/ct_faq.html
- ⁴⁹ Apache JMeter, <http://jakarta.apache.org/jmeter/>
- ⁵⁰ Waterfall model on Wikipedia, http://en.wikipedia.org/wiki/Waterfall_model
- ⁵¹ PHP: Hypertext Preprocessor, <http://php.net/>
- ⁵² MySQL AB :: The world's most popular open source database, <http://mysql.com/>
- ⁵³ Powered by phpBB © 2001, 2005 phpBB Group, <http://www.phpbb.com/>
- ⁵⁴ Powered by bBlog, <http://www.bblog.com/>
- ⁵⁵ Databaseteori – ordliste, <http://www.jbi.hio.no/bibin/KoG1db/2006/ressurser/ordliste.html>
- ⁵⁶ Topic Maps - framtidens standard for informasjonssøk, <http://www.abm-utvikling.no/publisert/synopsis/2000/nr-4/syn4-06.htm>